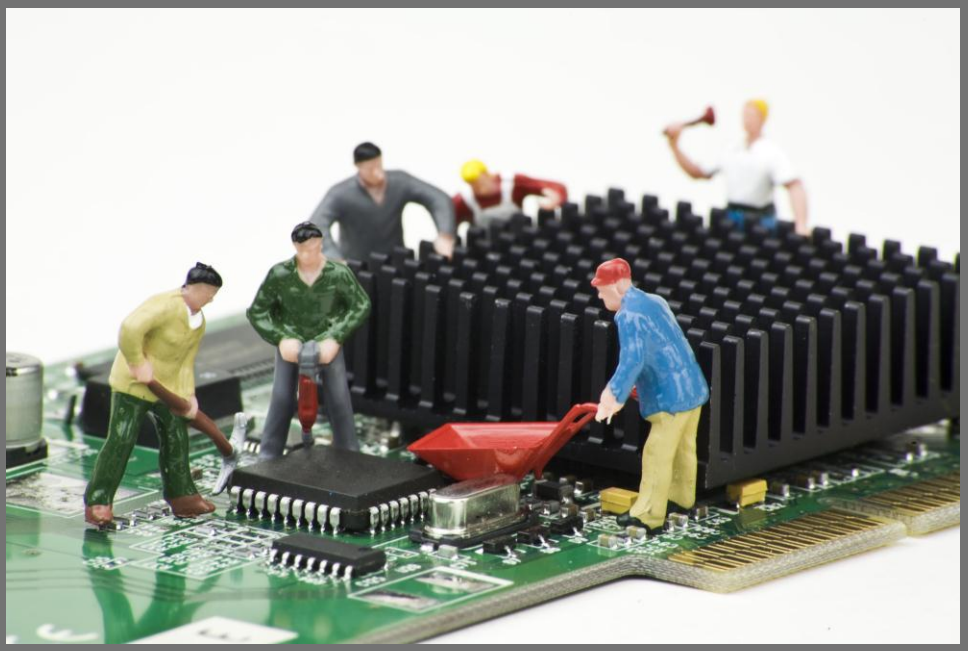
Modul Praktikum

DATAMINING



Migunani M.Kom

UNIVERSITAS SAINS DAN TEKNOLOGI
KOMPUTER SEMARANG

BAB I
FUNGSI DISKRIPSI DALAM DATAMINING

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan excel dan SPSS sebagai alat bantu data mining.
2. Mahasiswa dapat menjelaskan deskripsi grafis, lokasi dan keberagaman menggunakan perangkat lunak excel dan SPSS.

1.2. Pendahuluan

Datamining adalah disiplin ilmu yang tujuan utamanya adalah untuk menemukan, menggali atau menambang pengetahuan dari data atau informasi yang kita miliki. Data mining merupakan proses iteratif dan interaktif untuk menemukan pola atau model yang baru, bermanfaat, dan dimengerti dalam suatu database yang sangat besar (massive databases).

Datamining diperlukan karena beberapa alasan, diantaranya : ketersediaan data yang melimpah, kebutuhan akan informasi (pengetahuan) sebagai pendukung pengambilan keputusan untuk mendukung solusi bisnis, ketersediaan data transaksi dalam volume yang besar dan ketersediaan teknologi informasi yang terjangkau dan dapat diadopsi secara luas.

Ada berbagai cara dalam mendiskripsikan data yang besar dan beragam, serta pengetahuan yang dihasilkan, yaitu :

1. Deskripsi Grafis.
   Cara mendiskripsikan data dalam bentuk gambar umumnya berupa diagram Titik dan Histogram
2. Deskripsi Lokasi.
   Mendiskripsikan data untuk mengetahui posisi/tempat tertentu dari data, umumnya menggunakan nilai Rata-rata (mean), Nilai Tengah (Median), Sering Muncul (Modus), Empat Bagian (Kuartil), dan 100 Bagian (Persentil).
3. Deskripsi Keberagaman.
   Untuk mengetahui tingkat keberagaman data dapat menggunakan :
   - Nilai Rentang (Range) : Jarak antara data terkecil dengan data terbesar dari kelompok data tertentu.
   - Varians (S^2) : Jarak antara setiap data sampel dengan nilai pusatnya (rata-rata). Nilai Varians ini dapat digunakan untuk mengestimasi varians populasi (σ^2). Agar data memiliki satuan yang sama maka dibuatlah ukuran standar deviasi yang merupakan akar kuadrat varians. Misalnya data-data sampel memiliki satuan cm, maka varians memiliki satuan cm^2, karena standar deviasi merupakan akar kuadrat dari varians sehingga satuan data menjadi sama.

$$S^2 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})}{n-1}$$

   - Standar Deviasi (σ) : simpangan baku / nilai akar kuadrat dari Varians.

$$\sigma = \sqrt{S^2}$$

1.3. Pengolahan Data Menggunakan Excel

1. Buatlah tabel tinggi badan mahasiswa berikut ini menggunakan microsoft excel.

No	Tinggi Badan Biasa
1	168
2	164
3	167
4	164
5	171
6	166
7	169
8	172
9	166
10	166

No	Tinggi Badan Plus
1	175
2	176
3	183
4	180
5	177
6	177
7	182
8	179
9	179
10	171

No	Tinggi Badan Biasa (Urut)
1	164
2	164
3	166
4	166
5	166
6	167
7	168
8	169
9	171
10	172

2. Gunakan formula dalam excel untuk data diatas :
   a) Avarage (rata-rata) dengan formula = AVERAGE (range tinggi badan)
   b) Median (nilai tegah) dengan formula = MEDIAN (range tinggi badan)
   c) Modus (nilai sering) dengan formula = MODE (range tinggi badan)
   d) Quartil (empat bagian) dengan formula :
      - Quartil 1 => QUARTILE(range tinggi urut ;1)
      - Quartil 2 => QUARTILE(range tinggi urut ;2)
      - Quartil 3 => QUARTILE(range tinggi urut ;3)
      - Quartil 4 => QUARTILE(range tinggi urut ;4)
   e) Persentil (100 bagian) dengan formula (empat contoh) :
      - Persentasi 10 => PERCENTILE(range tinggi urut ; 0.10)
      - Persentasi 20 => PERCENTILE(range tinggi urut ; 0.25)
      - Persentasi 50 => PERCENTILE(range tinggi urut ; 0.50)
      - Persentasi 75 => PERCENTILE(range tinggi urut ; 0.75)
      - Persentasi 100 => PERCENTILE(range tinggi urut ; 0.100)
   f) Range, Varians dan Standar Deviasi dengan formula :
      - Nilai Range => MAX(range tinggi badan)-MIN(range tinggi badan)
      - Nilai Varians => VAR(range tinggi badan)
      - Standar Deviasi => STDEV(range tinggi badan)

1.4. Pengolahan Data Menggunakan SPSS

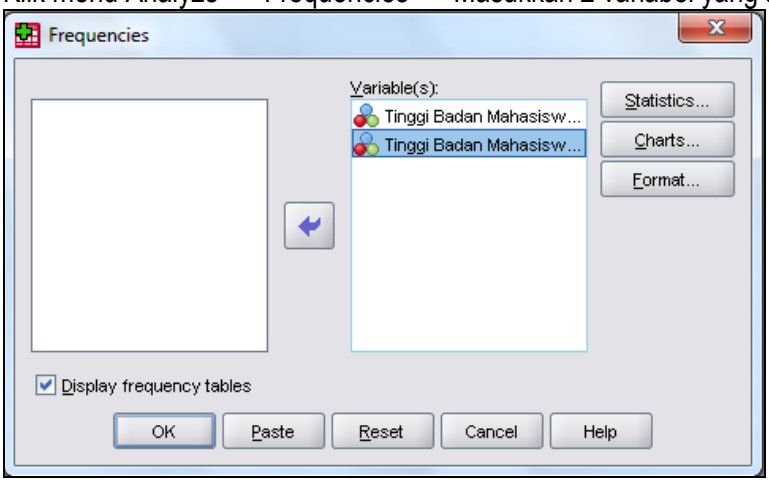
1. Buatlah tabel tinggi badan mahasiswa berikut ini menggunakan SPSS 16.
   Pada Tab Variabel View, buatlah Variabel berikut ini :

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
1	TinggiBadanBiasa	Numeric	3	0	Tinggi Badan Mahasiswa Biasa	None	None	15	Center	Nominal
2	TinggiBadanPlus	Numeric	3	0	Tinggi Badan Mahasiswa Plus	None	None	15	Center	Nominal

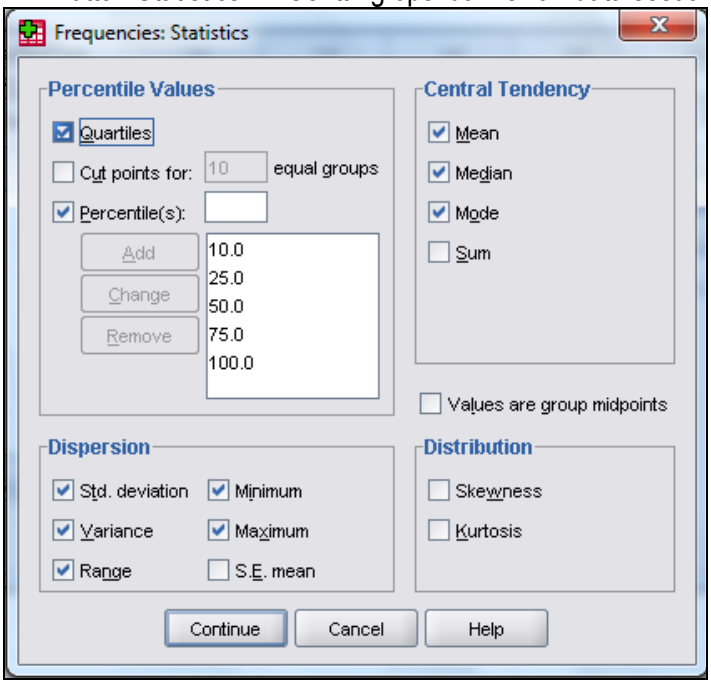
2. Selanjutnya isikan data berikut ini pada bagian Data View.

	TinggiBadanBiasa	TinggiBadanPlus
1	164	176
2	164	180
3	166	177
4	166	179
5	166	171
6	167	183
7	168	175
8	169	182
9	171	177
10	172	179

3. Untuk mendiskripsikan data diatas (statistik diskriptif), langkah-lagkahnya :
   - Klik menu Analyze >> Frequencies >> Masukkan 2 variabel yang ada.



   - Klik Button Statistics >> Centang opsi dan isikan data sesuai gambar berikut :



– Klik Button Continue >> kemudian klik Button OK, hasil pengolahan akan ditampilkan seperti gambar berikut :

Statistics

		Tinggi Badan Mahasiswa Biasa	Tinggi Badan Mahasiswa Plus
N	Valid	10	10
	Missing	0	0
Mean		167.30	177.90
Median		166.50	178.00
Mode		166	177[a]
Std. Deviation		2.710	3.510
Variance		7.344	12.322
Range		8	12
Minimum		164	171
Maximum		172	183
Percentiles	10	164.00	171.40
	25	165.50	175.75
	50	166.50	178.00
	75	169.50	180.50
	100	172.00	183.00

a. Multiple modes exist. The smallest value is shown

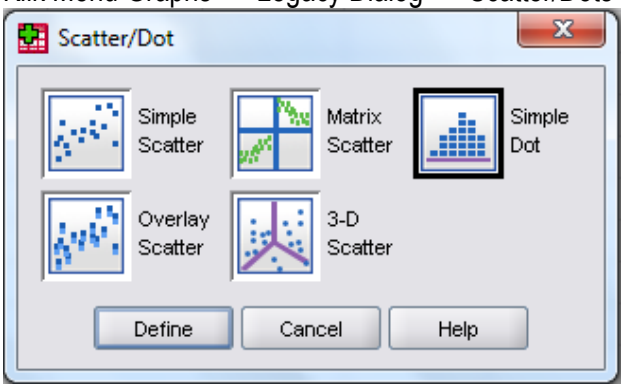
Tinggi Badan Mahasiswa Biasa

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	164	2	20.0	20.0	20.0
	166	3	30.0	30.0	50.0
	167	1	10.0	10.0	60.0
	168	1	10.0	10.0	70.0
	169	1	10.0	10.0	80.0
	171	1	10.0	10.0	90.0
	172	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

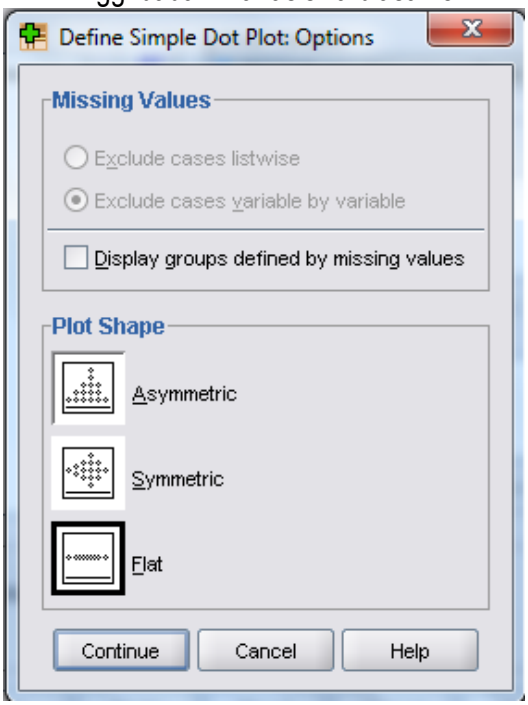
Tinggi Badan Mahasiswa Plus

		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	171	1	10.0	10.0	10.0
	175	1	10.0	10.0	20.0
	176	1	10.0	10.0	30.0
	177	2	20.0	20.0	50.0
	179	2	20.0	20.0	70.0
	180	1	10.0	10.0	80.0
	182	1	10.0	10.0	90.0
	183	1	10.0	10.0	100.0
	Total	10	100.0	100.0	

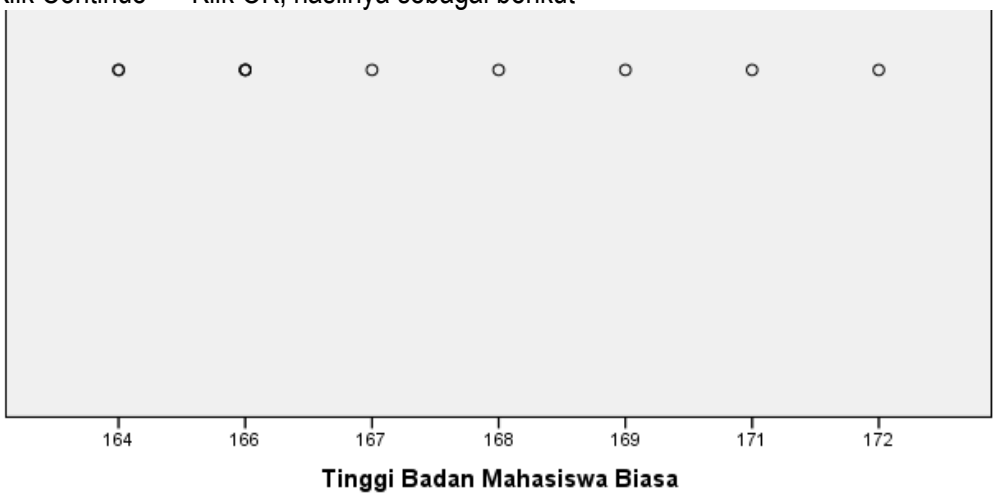
4. Untuk diskripsi secara grafis (titik/dot) dapat dilakukan dengan tahapan berikut :
   - Klik Menu Graphs >> Legacy Dialog >> Scatter/Dots >> Simple Dots



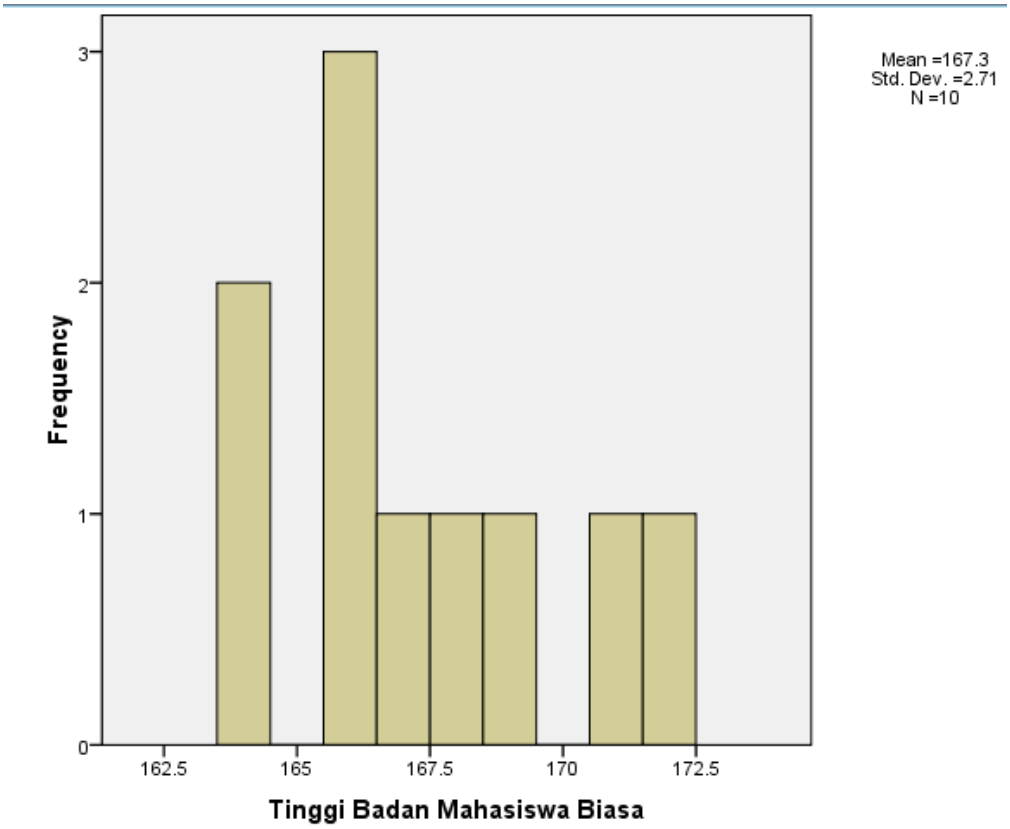
   - Pilih Tinggi badan mahasiswa biasa ke >> X-Axis Variabel >> Options >> Flat



   - Klik Continue >> Klik OK, hasilnya sebagai berikut



5. Untuk diskripsi secara grafis (histogram) dapat dilakukan dengan tahapan berikut :
   - Klik Menu Graphs >> Legacy Dialog >> Histogram >> Pilih variabel Tinggi badan mahasiswa biasa >> Klik OK.



BAB 2
FUNGSI ESTIMASI DALAM DATAMINING

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan excel dan SPSS sebagai alat bantu data mining.
2. Mahasiswa dapat menjelaskan estimasi titik dan estimasi selang menggunakan perangkat lunak SPSS.

1.2. Pendahuluan

Fungsi datamining berikutnya adalah estimasi yang bermanfaat untuk menentukan nilai estimasi terhadap susuatu. Cara estimasi terdiri dari dua bentuk yaitu estimasi titik dan selang kepercayaan. Estmimasi titik merupakan estimasi yang menghasilkan 1 buah nilai berupa angka. Mengingat ukuran populasi yang terus bertambah, maka penghitungan rata-rata dan standar deviasi atau varians akan selalu berulang. Berbekal sampel yang ada kita dapat melakukan estimasi (perkiraan) berdasarkan populasinya.

Estimasi titik merupakan bentuk estimasi yang menghasilkan satu buah nilai estimasi saja, yaitu berupa angka. Apa yang akan diperkirakan adalah sesuatu yang tidak kita ketahui nilai sebenarnya, yaitu karakteristik sebuah populasi. Rata-rata dan Varians merupakan besaran yang umum untuk menyatakan karakteristik populasi. Karakteristik Populasi biasa disebut dengan Parameter Populasi. Cara estimasi/memperkirakan 2 parameter pupulasi (varians dan rata-rata populasi) estimasi pada titik adalah :

1. Rata-rata populasi (μ) dapat diestimasi dengan rata-rata sampel ($\overline{x}$).
2. Varians Populasi (σ^2) dapat diestimasi dengan varians sampel (s^2).

Estimasi selang kepercayaan merupakan cara estimasi menggunakan rentang nilai dan selang kepercayaan karena kemungkinan nilai estimasi titik kurang memuaskan dan mungkin juga meleset (tidak tepat/error). Dibuatlah perkiraan lain berupa estimasi selang kepercayaan yang memiliki nilai batas bawah dan nilai batas atas. Adapun nilai batas bawah dan batas atas menggunakan rumus :

1. Batas bawah (L) = $\overline{X} - \mathbf{Z}_{\alpha/2}\sigma_{\overline{x}}$
2. Batas bawah (U) = $\overline{X} + \mathbf{Z}_{\alpha/2}\sigma_{\overline{x}}$
3. Nilai $\sigma_{\overline{x}} = \frac{\sigma}{\sqrt{n}}$

$\overline{X}$ = Rata-rata sampel
α = Selang Kepercayaan
Z = Nilai Distribusi Normal (lihat di tabel)
σ = Nilai Standar Deviasi
n = Jumlah Sample

1.3. Pengolahan Data Menggunakan Excel (Menghitung Estimasi-Titik)

1. Buatlah tabel volume 12 botol (xi) berikut ini menggunakan microsoft excel.

No	Xi (dalam ml)	$(x_i-\bar{x})^2$
1	2016	81
2	2025	324
3	1968	1521
4	2007	0
5	2031	576
6	2055	2304
7	2039	1024
8	1981	676
9	1975	1024
10	1964	1849
11	2036	841
12	1987	400
Rata-rata	**2007**	

Cel B4

Cel A15

Cel B16

2. Untuk menghitung $(x_i-\bar{x})^2$ yaitu, data sampel ke-i (misalnya data ke-1 = 2016) di kurangi dengan rata-rata semua data (yaitu 2007) kemudian di kuadratkan, maka akan diperoleh angka 81. Formula excelnya =(B4-B16)*(B4-B16)
3. Untuk data no.3 sampai no.12 Ulangi langkah pada no.2 dengan merubah cel dari B5 s/d B15.
4. Untuk memperoleh nilai Rata-rata populasi (μ) dapat diestimasi dengan rata-rata sampel ($\bar{x}$) dengan formula excel =AVERAGE(B4:B15)

$$\bar{x} = \sum_{i=1}^{n} \frac{xi}{n}$$

5. Untuk memperoleh nilai Varians Populasi (σ^2) dapat diestimasi dengan varians sampel (s^2) dengan formula excel =C16/(A15-1), dimana Cel C16 adalah jumlah total $(x_i-\bar{x})^2$ dengan formula excel =SUM(C4:C15).

$$s^2 = \sum_{i=1}^{n} \frac{(xi-\bar{x})^2}{(n-1)}$$

6. Agar satuan perhitungan sama maka dicari standar deviasi dengan formula excel =SQRT(Cel Nilai Variansya).

Pengetahuan yang diperoleh :
1. Pada umumnya setiap botol akan diisi air 2007 ml (rata-rata)
2. Keberagaman (jarak setiap data dengan pusat datanya) sebesar 965,45 ml^2 (varian)

1.4. Pengolahan Data Menggunakan Excel (Menghitung Estimasi-Selang)

1. Berdasarkan tabel yang sudah dibuat sebelumnya akan dicari dengan nilai selang kepercayaan (α) 95%, sehingga nilai $Z_{\alpha/2}$ (dibaca nilai distribusi normal selang kepercayaan dibagi dengan dua).

$\alpha/2$ = (100%-95%)/2
= 5% (0,025), sehingga nilai **$Z_{\alpha/2}$ =** -1.90+0.06 = -1.96

$$\sigma \bar{x} = \frac{\sigma}{\sqrt{n}} = \frac{31,7}{\sqrt{12}} = 8,97$$ => 31,7 => Standar Deviasi, n = Jumlah sampel

Formula Excel untuk $\sigma \bar{x} = \frac{\sigma}{\sqrt{n}}$ **=31.7/SQRT(12)**

a) Batas bawah (L) = $\bar{X} - Z_{\alpha/2}\sigma \bar{x}$

= 2007 + (-1.96 x 8,97) = 1898.42

Formula Excel untuk Batas bawah (L) **= 2007+(H7*H9)**

b) Batas bawah (U) = $\bar{X} + Z_{\alpha/2}\sigma \bar{x}$

= 2007 + (+1.96 x 8,97) = 2024.58

Formula Excel untuk Batas Atas (U) **= 2007- (H7*H9)**

1.5. Pengolahan Data Menggunakan SPSS (Menghitung Estimasi-Titik)

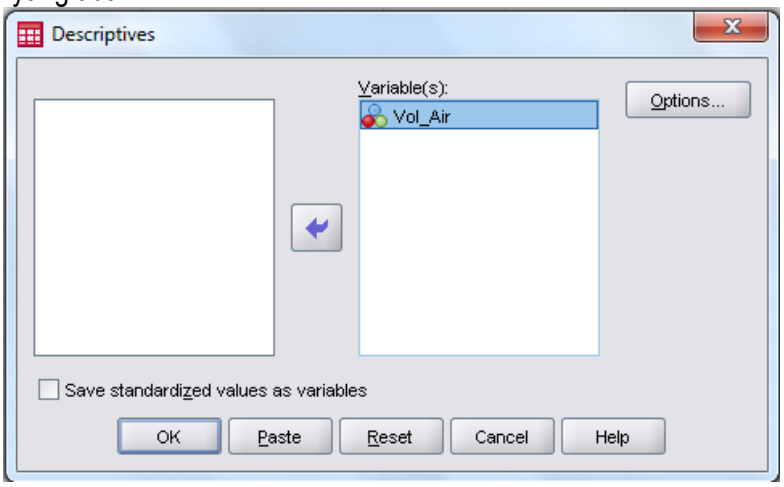
– Buatlah tabel tinggi volume air berikut ini menggunakan SPSS 16.
Pada Tab Variabel View, buatlah Variabel berikut ini

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
1	Vol_Air	Numeric	8	0		None	None	8	Right	Nominal

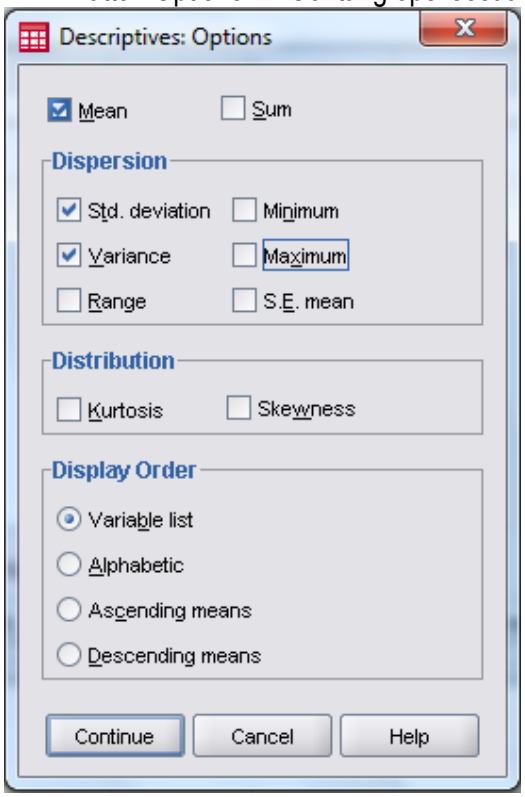
– Selanjutnya isikan data berikut ini pada bagian Data View.

	Vol_Air
1	2016
2	2025
3	1968
4	2007
5	2031
6	2055
7	2039
8	1981
9	1975
10	1964
11	2036
12	1987

- Untuk rata-rata populasi dengan estimasi titik (statistik diskriptif), langkah-lagkahnya :
  - Klik menu Analyze >> Descriptive Statistic >> Descriptives >> Masukkan 2 variabel yang ada.



  - Klik Button Options >> Centang opsi sesuai gambar berikut :



- Klik Button Continue >> kemudian klik Button OK, hasil pengolahan akan ditampilkan seperti gambar berikut :

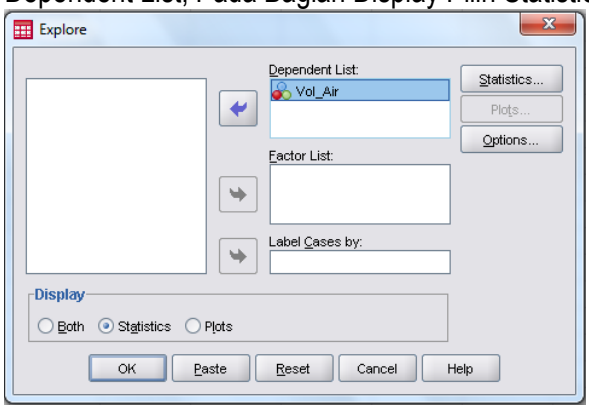
Descriptive Statistics

	N	Mean	Std. Deviation	Variance
Vol_Air	12	2007.00	31.072	965.455
Valid N (listwise)	12			

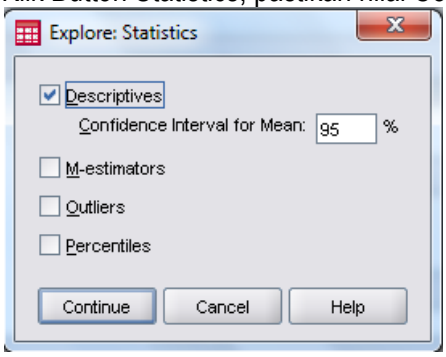
1.6. Pengolahan Data Menggunakan SPSS (Menghitung Estimasi-Selang)

– Untuk rata-rata populasi dengan estimasi selang (statistik diskriptif), langkah-langkahnya :

o Klik menu Analyze >> Descriptive Statistic >> Explore >> Masukkan variabel Vol_Air Ke Dependent List, Pada Bagian Display Pilih Statistics.



o Klik Button Statistics, pastikan nilai Convidence Interval 95% seperti gambar dibawah ini :



– Klik Button Continue >> kemudian klik Button OK, hasil pengolahan akan ditampilkan seperti gambar berikut :

Case Processing Summary

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
Vol_Air	12	100.0%	0	.0%	12	100.0%

Descriptives

			Statistic	Std. Error
Vol_Air	Mean		2007.00	8.970
	95% Confidence Interval for Mean	Lower Bound	1987.26	
		Upper Bound	2026.74	
	5% Trimmed Mean		2006.72	
	Median		2011.50	
	Variance		965.455	
	Std. Deviation		31.072	
	Minimum		1964	
	Maximum		2055	
	Range		91	
	Interquartile Range		58	
	Skewness		-.033	.637
	Kurtosis		-1.514	1.232

BAB 3
FUNGSI PREDIKSI DALAM DATAMINING

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan excel dan SPSS sebagai alat bantu data mining.
2. Mahasiswa dapat menjelaskan prediksi dengan regresi linier sederhana dan regresi linier berganda menggunakan perangkat lunak excel dan SPSS.

1.2. Pendahuluan

Fungsi estimasi pada pembahasan sebelumnya adalah kegiatan memperkirakan suatu hal, misalnya rata-rata populasi dari sejumlah sampel yang dimiliki. Sementara itu pada fungsi prediksi, data sampel digunakan untuk memprediksi hasil dari suatu hal baru yang akan segera muncul selanjutnya. Dapat disimpulkan bahwa estimasi dilakukan untuk memperkirakan hal yang tidak diketahui sedangkan prediksi memperkirakan hasil dari sesuatu hal yang belum terjadi.

Fungsi Prediksi Pada Datamining menggunakan Metode Regresi Linier. Ada Dua Macam Regresi Linier, yaitu Regresi Linier Sederhana dan Regresi Linier Berganda. Pada regresi linier sederhana hanya melibatkan satu variabel pemberi pengaruh, sementara regresi linier berganda melibatkan lebih dari satu variabel pemberi pengaruh. Variabel adalah besaran yang berubah-ubah nilainya.

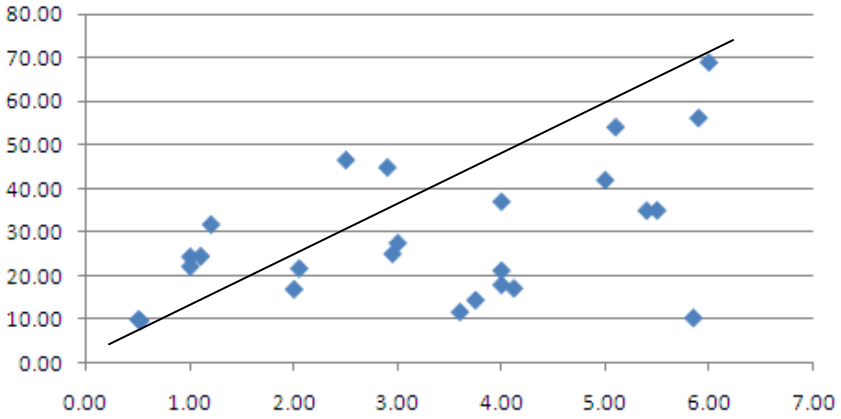
Variabel pemberi pengaruh dianalogikan sebagai sebab, variabel terpengaruh dianalogikan sebagai akibat. Misalnya Variabel yang dianggap relevan adalah variabel jarak rumah pelanggan dan waktu tempuh pengiriman pesanan. Kedua variabel tersebut dapat dipilah menjadi dua jenis, yaitu variabel pemberi pengaruh dan variabel terpengaruh. Secara logis jauh dekatnya rumah pelanggan mengakibatkan panjang pendeknya waktu pengiriman. Sehingga Jarak merupakan variabel yang memberi pengaruh, sedangkan waktu tempuh sebagai variabel terpengaruh.

1.3. Regresi Linier Sederhana

Regresi linier merupakan suatu cara memprediksi yang menggunakan garis lurus untuk menggambarkan hubungan diantara dua variabel atau lebih. Pada uraian kasus diatas terdapat dua variabel yaitu jarak dan waktu tempuh. Maka data akan digambarkan dengan sumbu X (Jarak) dan sumbu Y (Waktu Tempuh). Setiap pasangan data Jarak dan waktu tempuh digambarkan dengan titik. Sehingga tujuan Regresi Linier adalah untuk mencari sebuah garis lurus yang sedekat mungkin dengan semua titik (persamaan regresi), sehingga garis tersebut sah untuk mewakili semua titik. Secara umum persamaan garis regresi adalah :

$$Y = \beta_0 + \beta_{1x}$$

Y = Variabel Terpengaruh (Waktu) sebagai akibat
β_0 = Sebuah konstanta
β_1 = Gradient Garis
x = Variabel Pemberi Pengaruh (Jarak) sebagai akibat



Untuk mencari garis linier terbaik maka kita memerlukan konstanta dan gradienya. Gradien adalah perbandingan antara komponen y (ordinat) dan komponen x (absis) antara dua titik pada garis itu (dinotasikan m).

1.4. Regresi Linier Sederhana Menggunakan Excel

1. Buatlah tabel jarak dan waktu berikut ini menggunakan microsoft excel.

No	Jarak (X)	Waktu (Y)	Y_iX_i	X_i^2
1	0,50	9,95	4,98	0,25
2	1,10	24,45	26,90	1,21
3	1,20	31,75	38,10	1,44
4	5,50	35,00	192,50	30,25
5	2,95	25,02	73,81	8,70
6	2,00	16,86	33,72	4,00
7	3,75	14,38	53,93	14,06
8	0,52	9,60	4,99	0,27
9	1,00	24,35	24,35	1,00
10	3,00	27,50	82,50	9,00
11	4,12	17,08	70,37	16,97
12	4,00	37,00	148,00	16,00
13	5,00	41,95	209,75	25,00
14	3,60	11,66	41,98	12,96
15	2,05	21,65	44,38	4,20
16	4,00	17,89	71,56	16,00
17	6,00	69,00	414,00	36,00
18	5,85	10,30	60,26	34,22
19	5,40	34,93	188,62	29,16
20	2,50	46,59	116,48	6,25
21	2,90	44,88	130,15	8,41
22	5,10	54,12	276,01	26,01
23	5,90	56,23	331,76	34,81
24	1,00	22,13	22,13	1,00
25	4,00	21,15	84,60	16,00
Jumlah	**82,94**	**725,42**	**2745,81**	**353,18**
Rata-rata	**3,32**	**29,02**		

2. Hitunglah jumlah total semua data dengan fungsi SUM (misal sum=(B1:B25)) dan hitung pula rata-rata Jarak (x) dan Waktu (y) menggunakan average (misalnya AVERAGE (A1:B25)).
3. Hitunglah nilai β_1 (sesuaikan posisi cel hasil penghitungan jumlah dan rata-rata yang sudah dihitung sebelumnya (point 2)).

$$\beta_1 = \frac{\sum_{i=1}^{n} y_i x_i - \frac{\left(\sum_{i=1}^{n} y_i\right)\left(\sum_{i=1}^{n} x_i\right)}{n}}{\sum_{i=1}^{n} x_i^{\,2} - \frac{\left(\sum_{i=1}^{n} x_i\right)^2}{n}} = \frac{2745{,}81 - \frac{(752{,}42)(82{,}94)}{25}}{353{,}18 - \frac{(82{,}94)^2}{25}} = 4{,}35$$

4. Hitunglah nilai β_0 (sesuaikan posisi cel hasil penghitungan jumlah dan rata-rata yang sudah dihitung sebelumnya (point 2)).

$\beta_0 = \bar{y} - \beta_1 \bar{x}$ = 29,02 - 4,35 (3,32) = 14,60

5. Hitung nilai Variabel Y (sesuaikan posisi cel hasil penghitungan β_1 dan β_0 yang sudah dihitung sebelumnya (point 3 dan 4)).

$Y = \beta_0 + \beta_{1x}$ = 14.60 + 4.35 X

Pengetahuan Yang Diperoleh :
1. Kita dapat memprediksi bahwa waktu tempuh pengiriman pesanan sama dengan 14,60 menit + 4,35 kali jarak rumah pelanggan.
2. Persamaan garis ini menyatakan bahwa bila rumah pelanggan berjarak 0 km dari restoran waktu antarnya di prediksi 14,60 menit.
3. Jika jaraknya bertambah 1 km, maka waktu antar akan bertambah 4,35 menit menjadi 18,93 menit.

1.5. Regresi Linier Berganda

Persamaan regresi linier berganda tidak hanya melibatkan satu variabel pemberi pengaruh. Persamaan regresi dibangun dengan lebih dari satu variabel pemberi pengaruh. Apabila terdapat *k buah variabel pemberi* pengaruh, maka bentuk persamaan regresinya menjadi **Y adalah variabel terpengaruh**, **βo** adalah sebuah konstanta, **β1** adalah gradient pertama, **X1** adalah variabel pemberi pengaruh pertama, **β2** adalah gradient kedua, **X2** adalah variabel pemberi pengaruh yang kedua. Persamaan-persamaan yang ditunjukkan dapat digunakan untuk menemukan persamaan garis regresi. Perhatikan bahwa terdapat (k+1) persamaan, sementara variabel yang tidak diketahui adalah sebanyak (k+1), yaitu dari βo hingga βk.

1.6. Regresi Linier Berganda Menggunakan Excel

1. Buatlah tabel jarak, lampu dan waktu berikut ini menggunakan microsoft excel.

No	Lampu (X1)	Jarak (X2)	Waktu (Y)	X_{i1}^2	$X_{i1}X_{i2}$	$X_{i1}Y_i$	X_{i2}^2	$X_{i2}Y_i$
1	2,00	0,50	9,95	4,00	1,00	19,90	0,25	4,98
2	8,00	1,10	24,45	64,00	8,80	195,60	1,21	26,90
3	11,00	1,20	31,75	121,00	13,20	349,25	1,44	38,10

4	10,00	5,50	35,00	100,00	55,00	350,00	30,25	192,50
5	8,00	2,95	25,02	64,00	23,60	200,16	8,70	73,81
6	4,00	2,00	16,86	16,00	8,00	67,44	4,00	33,72
7	2,00	3,75	14,38	4,00	7,50	28,76	14,06	53,93
8	2,00	0,52	9,60	4,00	1,04	19,20	0,27	4,99
9	9,00	1,00	24,35	81,00	9,00	219,15	1,00	24,35
10	8,00	3,00	27,50	64,00	24,00	220,00	9,00	82,50
11	4,00	4,12	17,08	16,00	16,48	68,32	16,97	70,37
12	11,00	4,00	37,00	121,00	44,00	407,00	16,00	148,00
13	12,00	5,00	41,95	144,00	60,00	503,40	25,00	209,75
14	2,00	3,60	11,66	4,00	7,20	23,32	12,96	41,98
15	4,00	2,05	21,65	16,00	8,20	86,60	4,20	44,38
16	4,00	4,00	17,89	16,00	16,00	71,56	16,00	71,56
17	20,00	6,00	69,00	400,00	120,00	1380,00	36,00	414,00
18	1,00	5,85	10,30	1,00	5,85	10,30	34,22	60,26
19	10,00	5,40	34,93	100,00	54,00	349,30	29,16	188,62
20	15,00	2,50	46,59	225,00	37,50	698,85	6,25	116,48
21	15,00	2,90	44,88	225,00	43,50	673,20	8,41	130,15
22	16,00	5,10	54,12	256,00	81,60	865,92	26,01	276,01
23	17,00	5,90	56,23	289,00	100,30	955,91	34,81	331,76
24	6,00	1,00	22,13	36,00	6,00	132,78	1,00	22,13
25	5,00	4,00	21,15	25,00	20,00	105,75	16,00	84,60
Jumlah	**206,00**	**82,94**	**725,42**	**2396,00**	**771,77**	**8001,67**	**353,18**	**2745,81**
Rata-rata	**8,24**	**3,32**	**29,02**					

2. Hitunglah jumlah total semua data dengan fungsi SUM (misal sum=(B1:B25)) dan hitung pula rata-rata Lampu (x1) Jarak (x2) dan Waktu (y) menggunakan average (misalnya AVERAGE (A1:B25))
3. Hitunglah nilai β_1 (sesuaikan posisi cel hasil penghitungan jumlah dan rata-rata yang sudah dihitung sebelumnya (point 2)).

$$n\beta_0 + \beta_1 \sum_{i=1}^{n} x_{i1} + \beta_2 \sum_{i=1}^{n} x_{i2} + \ldots + \beta_k \sum_{i=1}^{n} x_{ik} = \boxed{\sum_{i=1}^{n} y_i}$$

$$\beta_0 \sum_{i=1}^{n} x_{i1} + \beta_1 \sum_{i=1}^{n} {x_{i1}}^2 + \beta_2 \sum_{i=1}^{n} x_{i1}x_{i2} + \ldots + \beta_k \sum_{i=1}^{n} x_{i1}x_{ik} = \boxed{\sum_{i=1}^{n} x_{i1}y_i}$$

$$\beta_0 \sum_{i=1}^{n} x_{ik} + \beta_1 \sum_{i=1}^{n} x_{i1}x_{ik} + \beta_2 \sum_{i=1}^{n} x_{i2}x_{ik} + \ldots + \beta_k \sum_{i=1}^{n} {x_{ik}}^2 = \boxed{\sum_{i=1}^{n} x_{ik}y_i}$$

$$Y = \beta_0 + \beta_1 + \beta_{2X2} + \ldots + \beta_{kxk}$$

Sehingga hasil perhitungan persamaannya menjadi (lihat hasil pada tabel):

$$25\beta o + \beta_1 + \beta_2(82{,}94) = 725{,}42$$

$$\beta o(206) + \beta_1(2396) + \beta_2(771{,}77) = 8001{,}67$$

$$\beta o(82{,}94) + \beta_1(771{,}77) + \beta_2(353{,}18) = 2745{,}81$$

Ketiga persamaan tersebut di selesaikan sehingga diperoleh :

$$\beta o = 2{,}31 \qquad \beta_1 = 2{,}74 \qquad \beta_2 = 1{,}24$$

$$Y = 2{,}31 + 2{,}74X_1 + 1{,}24X_2$$

Pengetahuan Yang Diperoleh :
1. Kita dapat memprediksi bahwa waktu tempuh pengiriman pesanan sama dengan 2,31 menit + 2,74 kali banyaknya lampu merah yang ditemui di sepanjang perjalanan dan ditambah 1,24 kali jarak rumah pelanggan.

1.7. Regresi Linier Sederhana Menggunakan SPSS

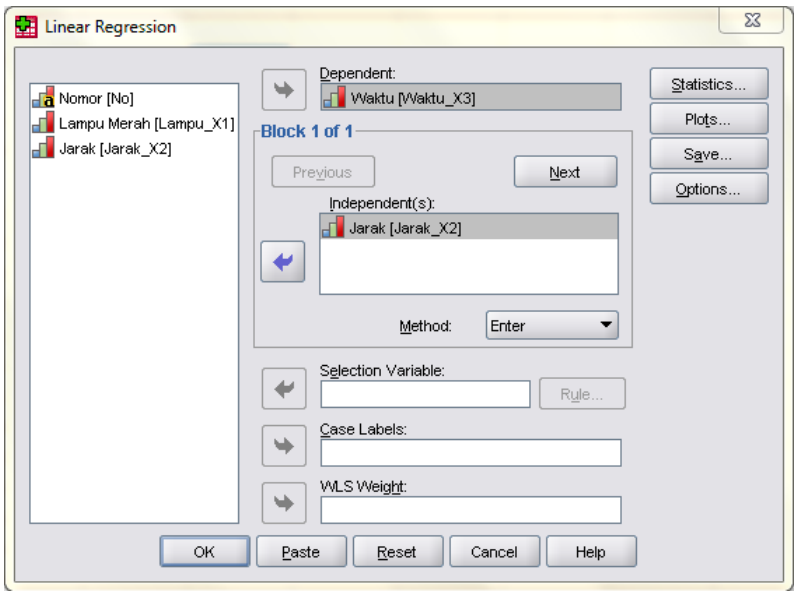
1. Buatlah tabel tinggi volume air berikut ini menggunakan SPSS 16. Pada Tab Variabel View, buatlah Variabel berikut ini.

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure
1	No	String	5	0	Nomor	None	None	8	Left	Ordinal
2	Lampu_X1	Numeric	5	2	Lampu Merah	None	None	10	Left	Ordinal
3	Jarak_X2	Numeric	5	2	Jarak	None	None	10	Left	Ordinal
4	Waktu_X3	Numeric	5	2	Waktu	None	None	9	Left	Ordinal

2. Selanjutnya isikan data berikut ini pada bagian Data View.

	No	Lampu_X1	Jarak_X2	Waktu_X3
1	1	2.00	0.50	9.95
2	2	8.00	1.10	24.45
3	3	11.00	1.20	31.75
4	4	10.00	5.50	35.00
5	5	8.00	2.95	25.02
6	6	4.00	2.00	16.86
7	7	2.00	3.75	14.38
8	8	2.00	0.52	9.60
9	9	9.00	1.00	24.35
10	10	8.00	3.00	27.50
11	11	4.00	4.12	17.08
12	12	11.00	4.00	37.00
13	13	12.00	5.00	41.95
14	14	2.00	3.60	11.66
15	15	4.00	2.05	21.65
16	16	4.00	4.00	17.89
17	17	20.00	6.00	69.00
18	18	1.00	5.85	10.30
19	19	10.00	5.40	34.93
20	20	15.00	2.50	46.59
21	21	15.00	2.90	44.88
22	22	16.00	5.10	54.12
23	23	17.00	5.90	56.23
24	24	6.00	1.00	22.13
25	25	5.00	4.00	21.15

3. Klik menu Analyze >> Regresion >> Linier, Pilih Waktu sebagai variabel Dependence dan Jarak Sebagai Variabel Independence.



4. Klik OK, perhatikan output hasil pengolahan seperti pada gambar berikut.

Model Summary[b]

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.492[a]	.242	.209	14.15715

a. Predictors: (Constant), Jarak

b. Dependent Variable: Waktu

Diperoleh nilai Adjusted R Square sebesar 0,209 artinya pengaruh jarak terhadap waktu sebesar 20,9%, sisanya (100%-20,9%)=79,1% adalah variabel bebas lain yang mempengaruhi waktu.

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1474.247	1	1474.247	7.356	.012[a]
	Residual	4609.774	23	200.425		
	Total	6084.021	24			

a. Predictors: (Constant), Jarak

b. Dependent Variable: Waktu

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	14.596	6.024		2.423	.024
	Jarak	4.347	1.603	.492	2.712	.012

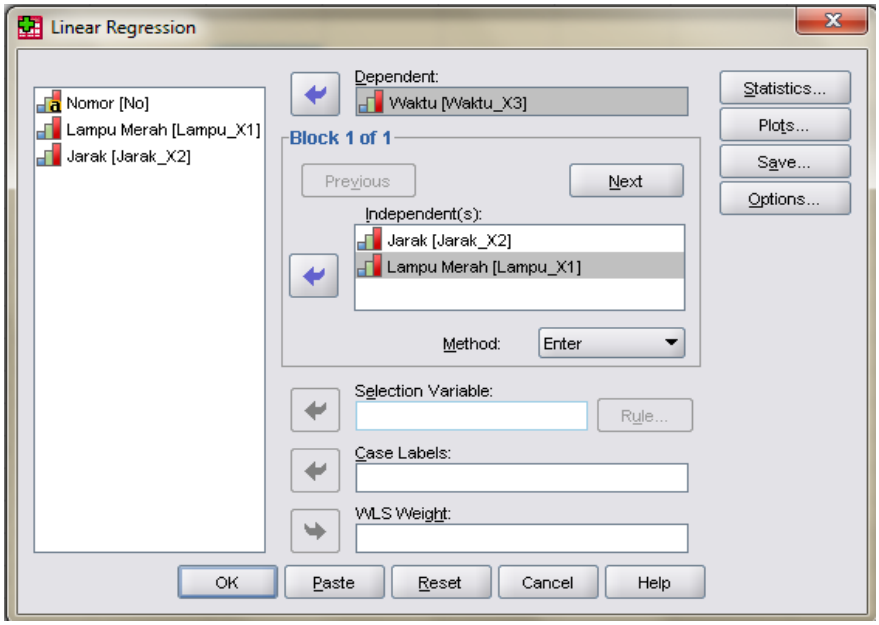
a. Dependent Variable: Waktu

Dapat disimpulkan bahwa nilai $\beta o = 14{,}596$ dan $\beta_1 = 4{,}347$ berdasarkan :

1) Uji t pada dasarnya menunjukkan seberapa jauh pengaruh satu variabel bebas secara individual dalam menerangkan variasi-variabel terikat. Tujuan dari uji t adalah untuk menguji koefisien regresi secara individual. Hasil uji t diperoleh nilai probabilitas (sig.) sebesar 0,12 (lebih besar dari nilai signifikansi 5% atau 0,05) sehingga dapat disimpulkan bahwa jarak tidak memenuhi signifikasi (ada variabel lain yang berpengaruh). Tidak memenuhi signifikansi artinya apabila jarak bertambah tidak secara signifikan akan menambah waktu tempuh demikian sebaliknya.
2) Tabel F dilakukan untuk mengetahui pengaruh variabel bebas secara bersama-sama terhadap variabel terikat. Hasil uji f diperoleh nilai probabilitas (sig.) sebesar 0,12 (lebih besar dari nilai signifikansi 5% atau 0,05) sehingga dapat disimpulkan bahwa jarak secara bersama-sama dengan variabel lain tidak secara signifikan berpengaruh terhadap waktu.

1.8. Regresi Linier Berganda Menggunakan SPSS

1. Klik menu Analyze >> Regresion >> Linier, Pilih Waktu sebagai variabel Dependence, sedangkan Jarak dan Lampu sebagai Variabel Independence.



2. Klik OK, perhatikan output hasil pengolahan seperti pada gambar berikut.

Variables Entered/Removed[b]

Model	Variables Entered	Variables Removed	Method
1	Lampu Merah, Jarak[a]	.	Enter

a. All requested variables entered.

b. Dependent Variable: Waktu

Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.991[a]	.981	.979	2.28679

a. Predictors: (Constant), Lampu Merah, Jarak

Variabel yang berpengaruh terhadap waktu tempuh adalah banyaknya lampu merah dan jarak tempuh. Diperoleh nilai Adjusted R Square sebesar 0,979 artinya pengaruh jarak terhadap waktu sebesar 97,9%, sisanya (100%-97,9%)=2,1% adalah variabel bebas lain yang mempengaruhi waktu tempuh.

ANOVA[b]

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	5968.974	2	2984.487	570.714	.000[a]
	Residual	115.047	22	5.229		
	Total	6084.021	24			

a. Predictors: (Constant), Lampu Merah, Jarak

b. Dependent Variable: Waktu

Coefficients[a]

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	2.309	1.059		2.180	.040
	Jarak	1.244	.280	.141	4.448	.000
	Lampu Merah	2.740	.093	.929	29.317	.000

a. Dependent Variable: Waktu

Dapat disimpulkan bahwa $\beta o = 2{,}31 \quad \beta_1 = 2{,}74 \quad \beta_2 = 1{,}24$ rdasarkan :

1) Hasil uji t diperoleh nilai probabilitas (sig.) sebesar 0,00 (lebih kecil dari nilai signifikansi 5% atau 0,05) sehingga dapat disimpulkan bahwa jarak dan waktu masing-masing memenuhi signifikansi artinya apabila jarak bertambah secara signifikan akan menambah waktu tempuh demikian pula sebaliknya.
2) Hasil uji f diperoleh nilai probabilitas (sig.) sebesar 0,00 (lebih kecil dari nilai signifikansi 5% atau 0,05) sehingga dapat disimpulkan bahwa jarak secara bersama-sama dengan variabel lampu merah secara signifikan berpengaruh terhadap waktu.

BAB 4
FUNGSI KLASIFIKASI DALAM DATAMINING

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan mechine learning WEKA 3.6.0 dan spreadsheet untuk klasifikasi.
2. Mahasiswa dapat menjelaskan klasifikasi menggunakan algoritma ID3 dan J48.

1.2. Pendahuluan

Klasifikasi adalah sebuah proses untuk menemukan model yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang kelasnya tidak diketahui (Tan *et all, 2004). Di dalam klasifikasi* diberikan sejumlah record yang dinamakan training set, yang terdiri dari beberapa atribut, atribut dapat berupa atribut kontinyu ataupun atribut kategoris, salah satu atribut menunjukkan kelas untuk suatu record. Model Klasifikasi terdiri dari :

1. Pemodelan Deskriptif, sebuah pemodelan dapat bertindak sebagai suatu alat yang bersifat menjelaskan untuk membedakan objek dengan kelas yang berbeda.
2. Pemodelan Prediktif, model klasifikasi ini menggunakan prediksi label kelas yang belum diketahui recordnya.

Tujuan dari klasifikasi adalah untuk menemukan model dari *training set yang membedakan record kedalam kategori atau* kelas yang sesuai, model tersebut kemudian digunakan untuk mengklasifikasikan *record yang kelasnya belum diketahui sebelumnya pada test set.* Selain itu klasifikasi bermanfaat untuk mengambil keputusan dengan memprediksikan suatu kasus, berdasarkan hasil klasifikasi yang diperoleh.

Untuk memperoleh model, harus dilakukan analisis terhadap data latih (*training set).* Sedangkan data uji (test set) digunakan untuk mengetahui tingkat akurasi dari model yang telah dihasilkan. Klasifikasi dapat digunakan untuk memprediksi nama atau nilai kelas dari suatu obyek data. Proses klasifikasi data dibedakan dalam 2 tahap :

1. Pembelajaran/Pembangunan Model
   Tiap – tiap record pada data latih dianalisis berdasarkan nilai – nilai atributnya, dengan menggunakan suatu algoritma klasifikasi untuk mendapatkan model.
2. Klasifikasi
   Pada tahap ini, data uji digunakan untuk mengetahui tingkat akurasi dari model yang dihasilkan. Jika tingkat akurasi yang diperoleh sesuai dengan nilai yang ditentukan, maka model tersebut dapat digunakan untuk mengklasifikasikan *record–record* baru yang belum pernah dilatihkan atau diujikan sebelumnya.

Untuk keperluan klasifikasi dalam mining data diperlukan kumpulan data untuk proses klasifikasi tersebut, data dalam klasifikasi :

a. Data dinyatakan dalam bentuk tabel yang memiliki atribut dan sejumlah record
b. Atribut menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan tree.
c. Salah satu atribut merupakan atribut yang menyatakan data solusi untuk setiap record data yang disebut dengan target atribut (class).

Proses yang terjadi pada algoritma Decision Tree adalah pengubahan data tabel menjadi model tree, mengubah model tree menjadi rules (aturan-aturan) dan menyederhanakan rule (pruning). Untuk menentukan nilai awal titik pada dicision tree perlu perhitungan entropi. Entropi merupakan suatu besaran yang digunakan untuk menentukan nilai awal yang akan dijadikan pembentukan tree. Pengertian lain Entroy (S) adalah jumlah bit yang diperkirakan dibutuhkan untuk dapat mengekstrak

suatu kelas (+atau -) dari sejumlah data acak pada ruang sampel S. Semakin kecil nilai entropy, maka semakin baik untuk digunakan dalam mengekstrak suatu kelas.

Formula entropi sebagai berikut : $Entropi(S) = -p.Log_2 p - q.Log_2 q$

Misalnya akan dicari aturan yang dapat digunakan untuk menentukan apakah seseorang menderita hipertensi atau tidak. Data yang diambil sebanyak 8 sampel dengan perkiraan bahwa yang mempengaruhi seseorang menderita hipertensi atau tidak adalah usia, berat badan dan jenis kelamin seperti pada tabel 1. Dimana masing-masing atribut yang diduga berpengaruhi tersebut memiliki nilai sebagai berikut :

1) Usia mempunyai instance Muda dan Tua
2) Berat Badan mempunyai instance Underweight, Average dan Overweight
3) Jenis Kelamin mempunyai instance Pria dan Wanita

Tabel 1. Data Pasien Kemungkinan Hipertensi

Nama	Usia	Berat	Kelamin	Hipertensi
Ali	Muda	Overweight	Pria	Ya
Adi	Muda	Underweight	Pria	Tidak
Ennie	Muda	Average	Wanita	Tidak
Budiman	Tua	Overweight	Pria	Tidak
Herman	Tua	Overweight	Pria	Ya
Didi	Muda	Underweight	Pria	Tidak
Rina	Tua	Overweight	Wanita	Ya
Gatot	Tua	Average	Pria	Tidak

Langkah-langkah penyelesian permasalahan diatas dengan Decision Tree adalah :

1) Menentukan Node Terpilih, untuk menentukan node terpilih, gunakan nilai Entropy dari setiap kriteria dengan data sampel yang ditentukan. Node terpilih adalah kriteria dengan entropy terkecil.

Usia	Hipertensi	Jumlah
Muda	Ya (+)	1
Muda	Tidak (-)	3
Tua	Ya (+)	2
Tua	Tidak (-)	2

Usia Muda :

$$q1 = -\frac{1}{4}Log_2\frac{1}{4} - \frac{3}{4}Log_2\frac{3}{4} = 0{,}81$$

Usia Tua :

$$q2 = -\frac{2}{4}Log_2\frac{2}{4} - \frac{2}{4}Log_2\frac{2}{4} = 1$$

Entrhopy Usia = $E = \frac{4}{8}q1 + \frac{4}{8}q2 = \frac{4}{8}(0{,}81) + \frac{4}{8}(1) = 0{,}91$

Kelamin	Hipertensi	Jumlah
Pria	Ya (+)	2
Pria	Tidak (-)	4
Wanita	Ya (+)	1
Wanita	Tidak (-)	1

Kelamin Pria :

$$q1 = -\frac{2}{6}Log_2\frac{2}{6} - \frac{4}{6}Log_2\frac{4}{6} = 0{,}92$$

Kelamin Wanita :

$$q2 = -\frac{1}{2}Log_2\frac{1}{2} - \frac{1}{2}Log_2\frac{1}{2} = 1{,}00$$

Entrhopy Kelamin = $E = \frac{6}{8}q1 + \frac{6}{8}q2 = \frac{2}{8}(0{,}81) + \frac{2}{8}(1) = 0{,}94$

Berat	Hipertensi	Jumlah
Overweignt	Ya (+)	3
Overweignt	Tidak (-)	1
Average	Ya (+)	0
Average	Tidak (-)	2
Underweight	Ya (+)	0
Underweight	Tidak (-)	2

Berat Overweight :

$$q1 = -\frac{3}{4}Log_2\frac{3}{4} - \frac{1}{4}Log_2\frac{1}{4} = 0{,}81$$

Berat Average :

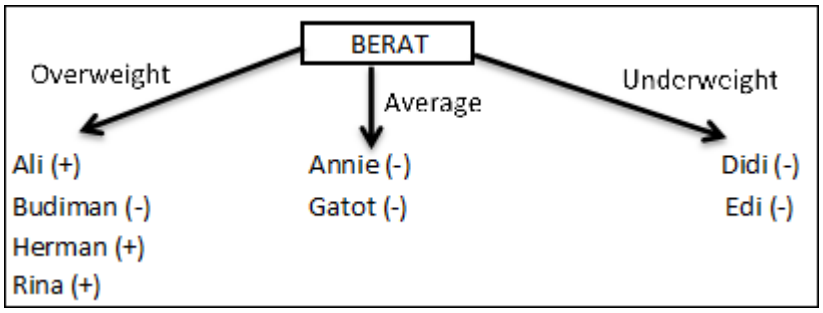
$$q2 = -\frac{0}{2}Log_2\frac{0}{2} - \frac{2}{2}Log_2\frac{2}{2} = 0{,}05$$

Berat Underwight :

$$q3 = -\frac{0}{2}Log_2\frac{0}{2} - \frac{2}{2}Log_2\frac{2}{2} = 0{,}05$$

Entrhopy Berat = $E = \frac{4}{8}q1 + \frac{2}{8}q2 + \frac{2}{8}q3 = \frac{4}{8}(0{,}81) + \frac{2}{8}(0{,}05) + \frac{2}{8}(0{,}05) = 0{,}43$

Atribut berat memiliki nilai Entrhopy terkecil yaitu **0,43**. Maka atribut berat menjadi node utama (*root node*) yang menghasilkan gambar berikut :



2) Selanjutnya akan di cari kandidat node berikutnya berdasarkan nilai Enthropy terkecil dari Usia dan Jenis Kelamin dan berada pada leaf yang memiliki nilai (+) dan (-).

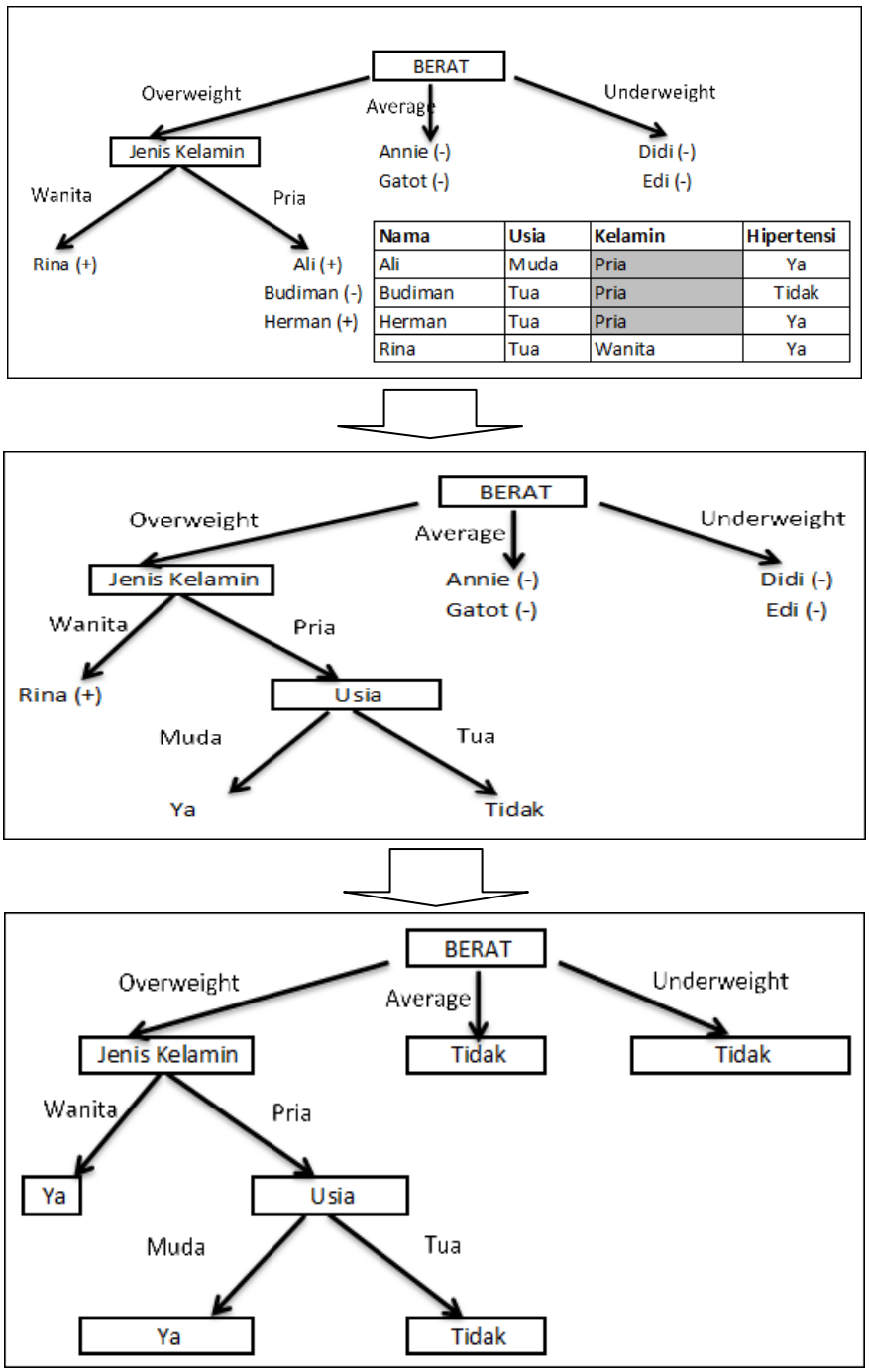
Nama	**Usia**	**Berat**	**Kelamin**	**Hipertensi**
Tini	Muda	Overweight	Pria	Ya
Tanti	Tua	Overweight	Pria	Tidak
Dina	Tua	Overweight	Pria	Ya
Rini	Tua	Overweight	Wanita	Ya

Usia	Hipertensi	Jumlah
Muda	Ya	1
	Tidak	0
Tua	Ya	2
	Tidak	1

Kelamin	Hipertensi	Jumlah
Pria	Ya	2
	Tidak	1
Wanita	Ya	1
	Tidak	0

Hasil perhitungan :
1. Enthropy Usia 0,69
2. Enthropy Kelamin 0,69

Dalam kondisi ini dapat meminta saran pakar untuk menentukan node berikutnya, misalnya ditentukan node berikutnya adalah atribut jenis kelamin kemudian atribut **usia**.



Nama	Usia	Kelamin	Hipertensi
Ali	Muda	Pria	Ya
Budiman	Tua	Pria	Tidak
Herman	Tua	Pria	Ya
Rina	Tua	Wanita	Ya

1.3. Menghitung enthropy menggunakan excel untuk menentukan node awal.

Buatlah tabel berikut ini (sesuaikan posisi cel untuk memudahkan perhitungan), masukkan formula untuk cel G5, G6, G7.

1. Enthropy Usia

Usia	Hipertensi	Jumlah	
Muda	Ya (+)	1	→ Cel D4
Muda	Tidak (-)	3	
Tua	Ya (+)	2	
Tua	Tidak (-)	2	→ Cel D7

Perhitungan

Enthropy Usia	Operasi	
	Cel	Angka
Usia Muda	0.81	0.81 → Cel G5
Usia Tua	1.00	1.00 → Cel G6
Enthropy Usia	**0.91**	**0.91** → Cel G7

Cel G5 =-(D4/(D4+D5))*LOG(D4/(D4+D5),2)-(D5/(D4+D5))*LOG(D5/(D4+D5),2)

Cel G6 =-(D6/(D6+D7))*LOG(D6/(D6+D7),2)-(D7/(D6+D7))*LOG(D7/(D6+D7),2)

Cel G7 =((D4+D5)/SUM(D4:D7))*G5+((D6+D7)/SUM(D4:D7))*G6

2. Enthropy Kelamin

Kelamin	Hipertensi	Jumlah	
Pria	Ya (+)	2	→ Cel L4
Pria	Tidak (-)	4	
Wanita	Ya (+)	1	
Wanita	Tidak (-)	1	→ Cel L7

Perhitungan

Enthropy Kelamin	Operasi	
	Cel	Angka
Pria	0.92	0.92 → Cel O5
Wanita	1.00	1.00 → Cel O6
Enthropy Kelamin	**0.94**	**0.94** → Cel O7

Cel O12 =-(L4/(L4+L5))*LOG(L4/(L4+L5),2)-(L5/(L4+L5))*LOG(L5/(L4+L5),2)
Cel O13 =-(L6/(L6+L7))*LOG(L6/(L6+L7),2)-(L7/(L6+L7))*LOG(L7/(L6+L7),2)
Cel O14 =((L4+L5)/SUM(L4:L7))*O5+((L6+L7)/SUM(L4:L7))*O6

3. Enthropy Berat

Berat	Hipertensi	Jumlah	
Overweignt	Ya (+)	3	→ Cel D19
Overweignt	Tidak (-)	1	
Average	Ya (+)	0	
Average	Tidak (-)	2	
Underweight	Ya (+)	0	
Underweight	Tidak (-)	2	→ Cel D24

Perhitungan

Enthropy Berat	Operasi	
	Cel	Angka
Overweight	0.81	0.81 → Cel G20
Average	0.05	0.50 → Cel G21
Underweight	0.05	0.50 → Cel G22
Enthropy Berat	**0.43**	**0.43** → Cel G23

Cel G20 =-(D19/(D19+D20))*LOG(D19/(D19+D20),2)-(D20/(D19+D20))*LOG(D20/(D19+D20),2)
Cel G21 =-(D21/(D21+D22))*LOG(D21/(D21+D22),2)-(D22/(D21+D22))*LOG(D22/(D21+D22),2)

Cel G22 =-(D23/(D23+D24))*LOG(D23/(D23+D24),2)-(D24/(D23+D24))*LOG(D24/(D23+D24),2)
Cel G23 =((D19+D20)/SUM(D19:D24))*G20+((D21+D22)/
SUM(D19:D24))*G21+((D23+D24)/SUM(D19:D24))*G22

Atribut berat memiliki nilai Enthropy terkecil yaitu **0,43**. Maka atribut berat menjadi node utama (*root node*). Dengan cara yang sama dapat dicari kandidat node berikutnya (kedua dan seterusnya), apakah atribut usia atau jenis kelamin (lihat langkah 2 hal 24).

1.4. Perangkat Lunak WEKA

WEKA merupakan perangkat lunak mesin pembelajaran (*tools machine learning*). WEKA merupakan kepanjangan dari *Waikato Environment for Knowledge Analysis*, dari Universitas Waikato, New Zealand untuk keperluan pendidikan dan penelitian. WEKA mampu menyelesaikan masalah-masalah *data mining* di dunia-nyata, khususnya klasifikasi yang mendasari pendekatan-pendekatan *machine learning*. Perangkat lunak ini ditulis dalam hirarki *class* Java dengan metode berorientasi objek dan dapat berjalan hampir di semua *platform*.

WEKA mudah digunakan dan diterapkan pada beberapa tingkatan yang berbeda. Tersedia implementasi algoritma-algoritma pembelajaran *state-of-the-art* yang dapat diterapkan pada dataset dari *command line*. WEKA mengandung *tools* untuk *pre-processing* data, klasifikasi, regresi, *clustering,* aturan asosiasi, dan visualisasi. *User* dapat melakukan *preprocess* pada data, memasukkannya dalam sebuah skema pembelajaran, dan menganalisa *classifier* yang dihasilkan dan performansinya – semua itu tanpa menulis kode program sama sekali. Contoh penggunaan WEKA adalah dengan menerapkan sebuah metode pembelajaran ke dataset dan menganalisa hasilnya untuk memperoleh informasi tentang data, atau menerapkan beberapa metode dan membandingkan performansinya untuk dipilih.

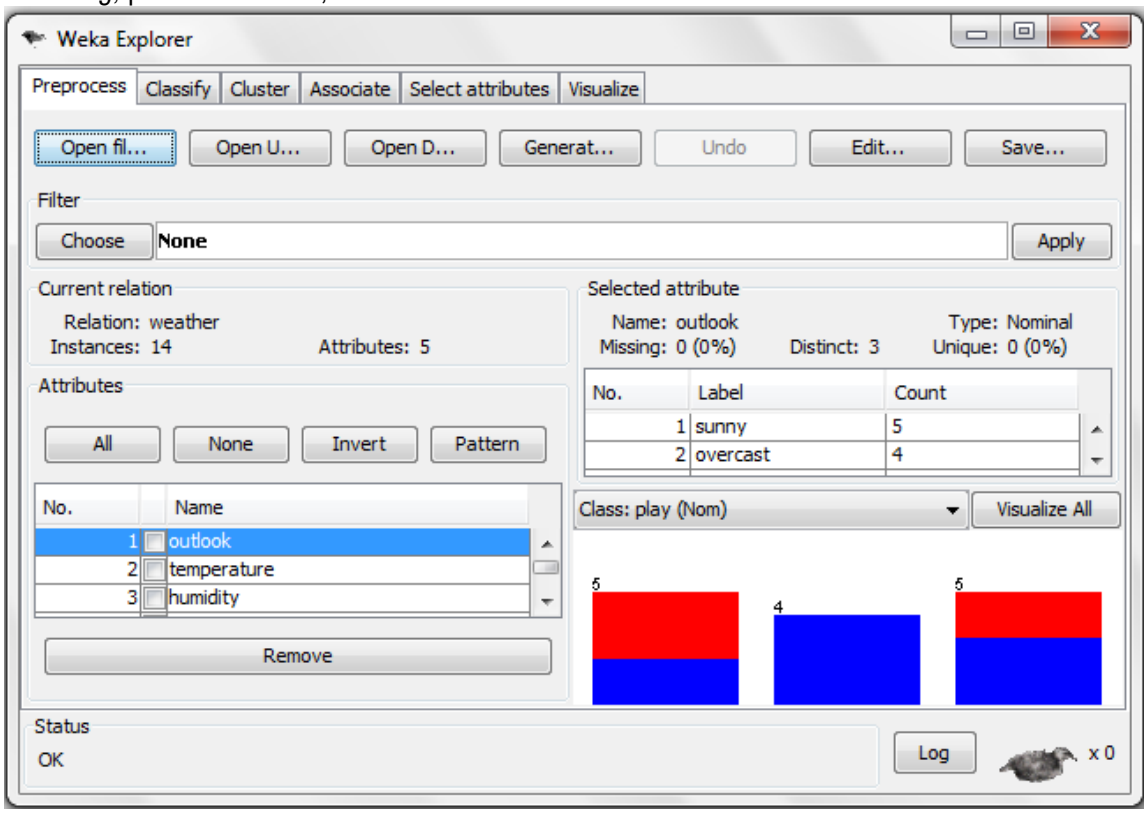
Tools yang dapat digunakan untuk *pre-processing* dataset membuat *user* dapat berfokus pada algoritma yang digunakan tanpa terlalu memperhatikan detail seperti pembacaan data dari file-file, implementasi algoritma *filtering*, dan penyediaan kode untuk evaluasi hasil. Pengembangan WEKA mengikuti model *releases* Linux: digit kedua yang genap menunjukkan *release* yang stabil dan digit kedua yang ganjil menunjukkan *release* 'pengembangan' (misalnya 3.0.x adalah *release* stabil, sedangkan 3.1.x adalah *release* yang sedang dikembangkan). Beberapa versi awal dari WEKA:

- WEKA 3.0 : versi buku, sesuai dengan deskripsi buku data mining.
- WEKA 3.2 : versi GUI, dengan penambahan GUI dari CLI.
- WEKA 3.3 : versi pengembangan dengan berbagai peningkatan.
- WEKA 3.4, 3.6 dan 3.7 : versi pengembangan berikutnya.

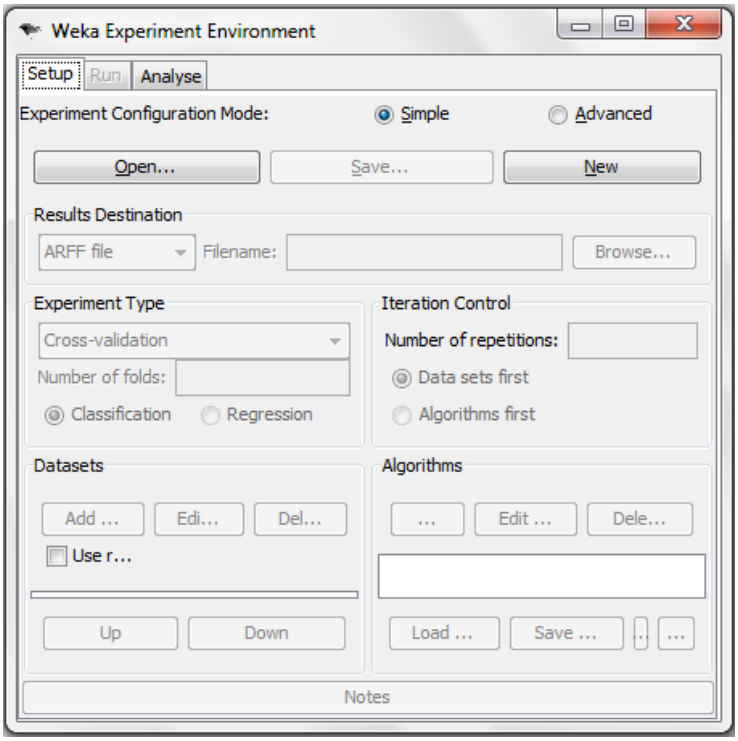
1.5. WEKA GUI Chooser

WEKA *GUI Chooser* adalah tampilan utama yang akan dilihat *user* pada saat pertama kali membuka perangkat lunak WEKA. Tampilan utama tersebut memberikan 4 pilihan GUI WEKA, yaitu *Simple CLI, Experimenter, Explorer,* dan *Knowledge Flow*.

1) **GUI Simple CLI**
   Merupakan GUI yang memungkinkan *user* mengetikkan perintah-perintah melalui *command line* menurut standar penggunaan *classifiers* maupun *filters*.
2) **GUI Explorer**
   GUI WEKA yang paling mudah digunakan dan menyediakan semua fitur WEKA dalam bentuk tombol dan tampilan visualisasi yang menarik dan lengkap. *Preprocess,* klasifikasi, asosiasi, *clustering,* pemilihan atribut, dan visualisasi.



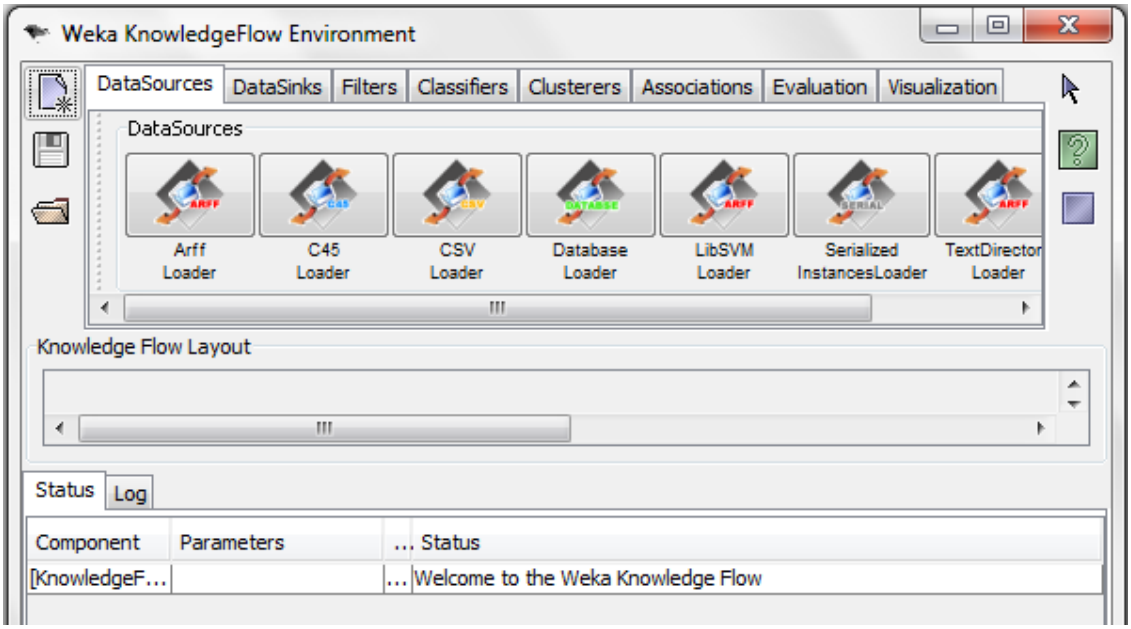
3) **GUI Experimenter**
   Memudahkan perbandingan performansi skema-skema pembelajaran yang berbeda. Experimenter biasanya digunakan untuk klasifikasi dan regresi. Hasil dari perbandingan performansi dapat dituliskan dalam file atau basis data. Pilihan evaluasi yang tersedia dalam WEKA adalah *cross-validation*, *learning curve*, *hold-out. User* juga dapat melakukan iterasi menurut beberapa setting parameter yang berbeda.
   a) Tab *Setup* yang muncul saat *user* membuka *Experimenter* memungkinkan *user* memilih dan mengkonfigurasi eksperimen yang dilakukan. Setelah menyimpan definisi eksperimen yang dilakukan, *user* dapat memulai eksperimen dari tab *Run* dan meng-klik tombol *Start*. Area di bawahnya akan menunjukkan proses yang sedang dilakukan. Hasilnya disimpan dalam format CSV dan dapat dibuka dalam bentuk *spreadsheet.*



b) Tab *Analize,* dapat digunakan untuk menganalisa hasil ekperimen yang dikirim ke WEKA. Jumlah baris hasil ditunjukkan pada panel *Source.* Hasilnya dapat di-*load* dalam format .ARFF maupun dari basis data.
Antarmuka ini memungkinkan *user* melakukan lebih dari 1 eksperimen sekaligus, mungkin menerapkan beberapa teknik berbeda pada sebuah dataset, atau teknik yang sama dengan parameter-parameter yang berbeda.

4) GUI Knowledge Flow

Merupakan GUI baru dalam WEKA yang merupakan antarmuka *Java-Beans-based* untuk melakukan *setting* dan menjalankan percobaan-percobaan *machine learning.*



Dalam GUI *Experimenter* ini, beberapa sumber data, *classifier*, dll dapat dihubungkan secara grafis. *User* juga dapat menggambarkan aliran data melalui komponen-komponen, misalnya : Data Source >> Filter >> Classifier >> Evaluator

KnowledgeFlow menyediakan alternatif lain dari *Explorer* sebagai sebuah *front end* grafis untuk algoritma-algoritma inti WEKA. Karena masih dalam pengembangan, beberapa fungsionalitas dalam *Explorer* belum tersedia dalam *KnowledgeFlow*.

KnowledgeFlow menampilkan aliran data dalam WEKA. *User* dapat memilih komponen-komponen WEKA dari *toolbar,* meletakkannya pada area yang tersedia dan menghubungkannya untuk membentuk aliran pengetahuan pemrosesan dan analisa data. *KnowledgeFlow* dapat menangani data secara *incremental* maupun dalam *batches* (*Explorer* hanya menangani data *batch*). Tentunya pembelajaran dari data secara *incremental* memerlukan sebuah *classifier* yang dapat diupdate *instance* per *instance*. Dalam WEKA tersedia 5 *classifiers* yang dapat menangani data secara *incremental*: *NaiveBayesUpdateable, IB1, IBk, LWR (Locally Weighted Regression)*. Tersedia pula sebuah *metadata classifier – RacedIncrementalLogitBoost* – yang dapat digunakan dari berbagai basis regresi untuk data *class* diskrit secara incremental.

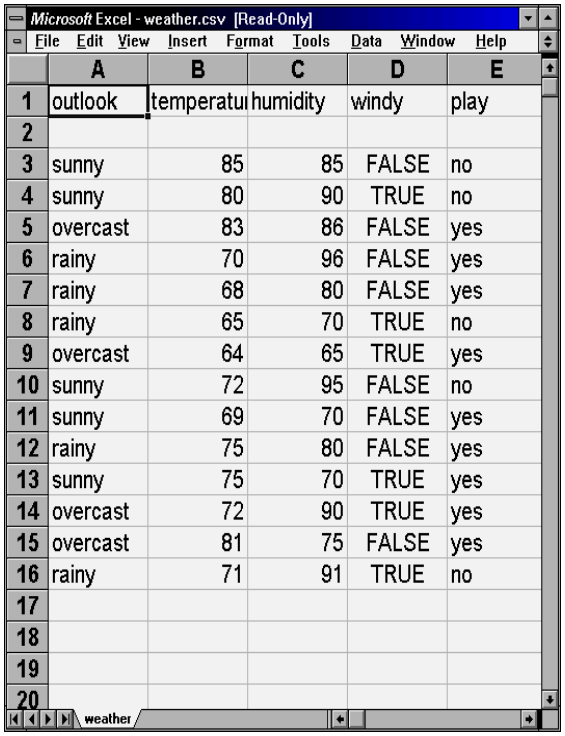
5) Format Data dalam WEKA

Misalnya diketahui sekumpulan data dan ingin dibangun sebuah *decision tree* dari data tersebut, maka data tersebut harus disimpan dalam format 'flat', ARFF karena WEKA perlu mengetahui beberapa informasi tentang tiap atribut yang tidak dapat disimpulkan secara otomatis dari nilai-nilainya.

File ARFF (*Attribute-Relation File Format*) adalah sebuah file teks ASCII yang berisi daftar *instances* dalam sekumpulan atribut. File ARFF dikembangkan oleh *Machine Learning Project* di *Department of Computer Science of The University of Waikato* untuk digunakan dalam perangkat lunak WEKA.

Pengubahan format data ini dapat dilakukan dengan mudah. Misalkan data awal dalam format .xls (lihat gambar 2a), buka data tersebut dari Microsoft Excel dan simpan sebagai .csv. Selanjutnya, buka file tersebut dari Microsoft Word, notepad, atau editor teks lainnya dan data sudah berubah dalam format *comma-separated value*. Lalu sesuaikan data tersebut dengan menambahkan informasi awal (gambar 2b). Hasilnya, data tersebut sudah dapat digunakan sebagai inputan dalam WEKA.
Pastikan bahwa data dalam format .arff tersebut sudah memenuhi kriteria :

- Data dipisahkan dengan koma, dengan kelas sebagai atribut terakhir.
- Bagian *header* diawali dengan @RELATION.
- Tiap atribut ditandai dengan @ATTRIBUTE. Tipe-tipe data dalam WEKA: numerik(REAL atau INTEGER), nominal, String, dan *Date*.
- Bagian data diawali dengan @DATA



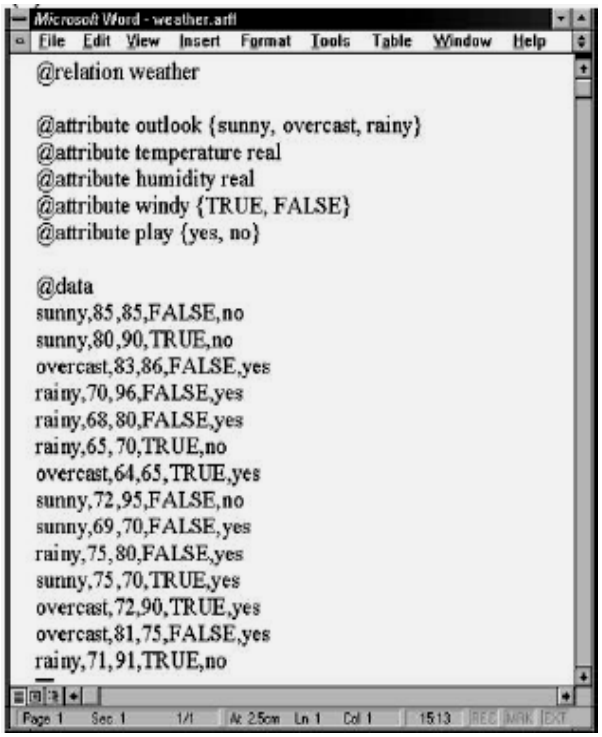
Microsoft Excel - weather.csv [Read-Only]

File Edit View Insert Format Tools Data Window Help

	A	B	C	D	E
1	outlook	temperatu	humidity	windy	play
2					
3	sunny	85	85	FALSE	no
4	sunny	80	90	TRUE	no
5	overcast	83	86	FALSE	yes
6	rainy	70	96	FALSE	yes
7	rainy	68	80	FALSE	yes
8	rainy	65	70	TRUE	no
9	overcast	64	65	TRUE	yes
10	sunny	72	95	FALSE	no
11	sunny	69	70	FALSE	yes
12	rainy	75	80	FALSE	yes
13	sunny	75	70	TRUE	yes
14	overcast	72	90	TRUE	yes
15	overcast	81	75	FALSE	yes
16	rainy	71	91	TRUE	no
17					
18					
19					
20					

weather

Gambar 2a. Format .csv



Microsoft Word - weather.arff

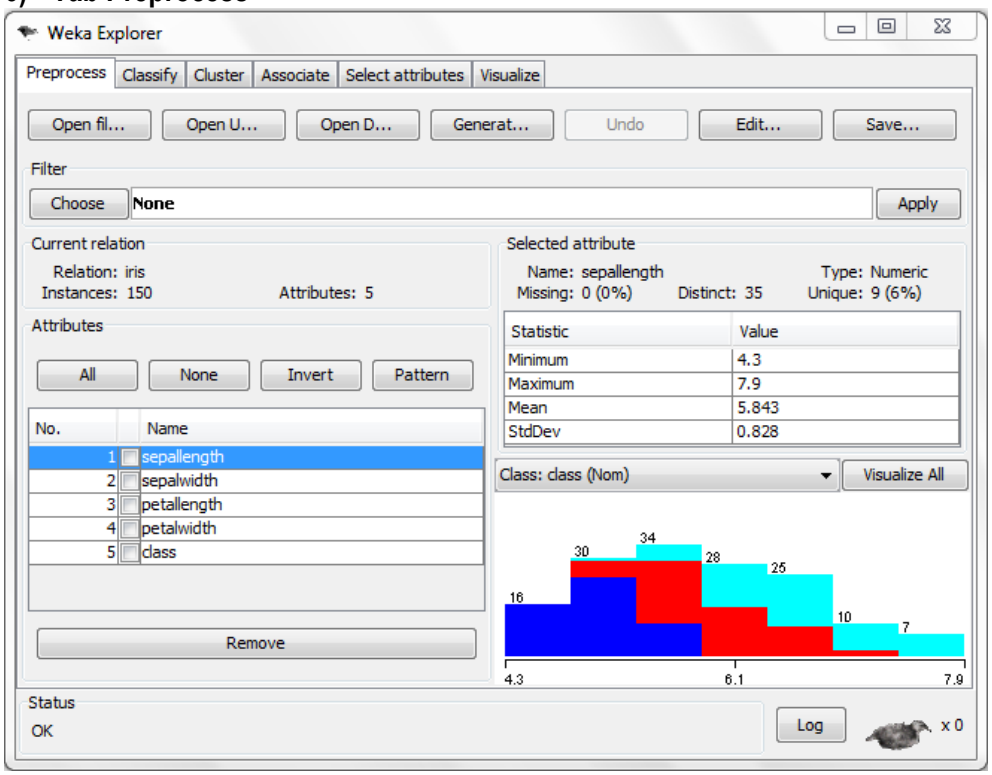
File Edit View Insert Format Tools Table Window Help

```
@relation weather

@attribute outlook {sunny, overcast, rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

Gambar 2b. Format .arff

WEKA Knowlege Explorer adalah sebuah *Graphical User Interface* (GUI) yang mudah digunakan dalam WEKA. Tiap paket utama WEKA (*Preprocess, Classify, Cluster, Associate,* dan *Select Attributes*) ditampilkan bersama perangkat *Visualization* yang memungkinkan himpunan data *Classifiers* dan *Clusterers* divisualisasikan dalam 2 dimensi.

6) Tab Preprocess



Gambar di atas menunjukkan tampilan tab *Preprocess* setelah *load* sebuah dataset (Iris.arff) dengan 150 *instances* dan 5 atribut, yaitu spallength, spallwidth, petallength, petallwidth dan class. Pada bagian kanan terdapat *selected attribute*, hal ini menunjukkan keterangan yang lebih terhadap attribut yang kita pilih berdasarkan tipe data yang ada seperti real, boolean(yes, no) atau sesuai dengan kriteria yang dinginkan oleh user. Misal, bila kita pilih attribute spallength, maka keterangan yang muncul adalah :

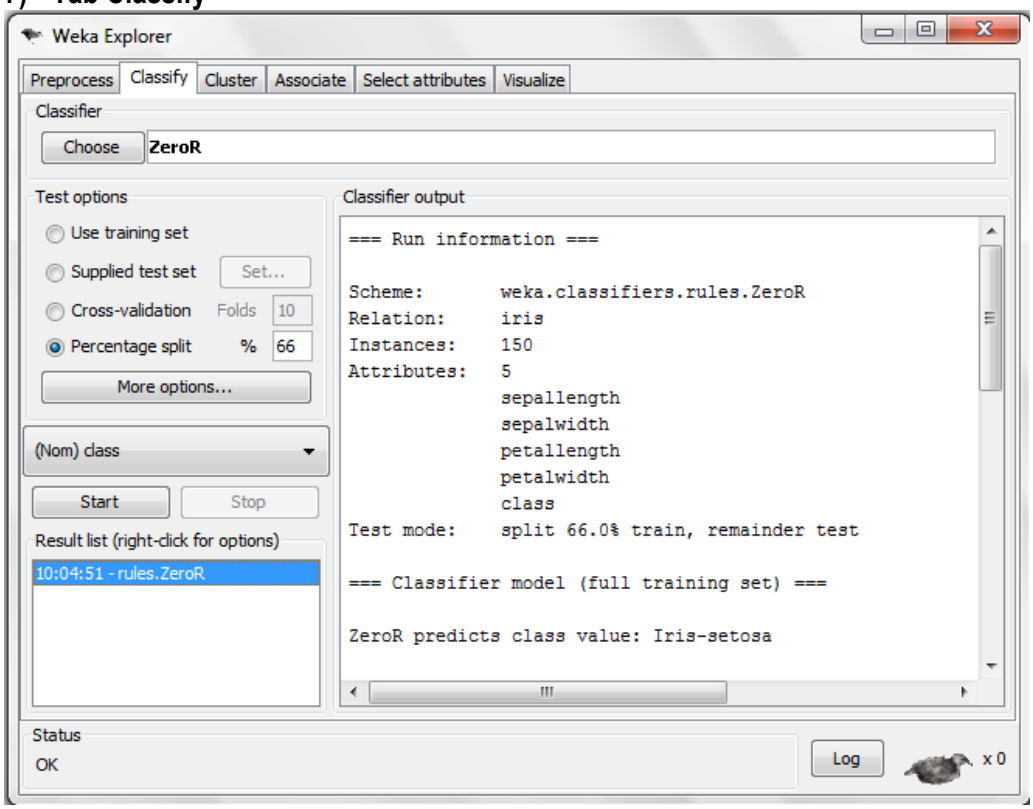
a. Nilai minimum : 4.3
b. Nilai maksimum : 7.9
c. Rata – rata (Mean) :5.843
d. Standar deviasinya (StdDev) : 0.828

Hal tersebut muncul karena attribute spallength mempunyai tipe data real. Begitupun bila terdapat atribute rasa (manis, pahit, asin), maka 3 tipe rasa itu akan muncul berapa jumlah yang ada pada datanya.

Visualisasi tiap atribut dapat dilihat dengan meng-klik tombol *Visualize.* Visualisasi ini menggunakan diagram batang, yang mengilustrasikan jumlah dari masing-masing tipe pada atribut yang ada. Seperti pada atribut *class* terdapat tiga tipe yaitu iris-sentosa, iris-versicolor dan iris-virgina, di visulisasikan dengan diagram batang yang mempunyai jumlah nilai yang sama yaitu 50.

Pada tab ini *user* dapat menentukan filter *unsupervised* yang akan diterapkan pada data. Filter berperan dalam mengubah data dengan berbagai cara. Klik pada filter tertentu yang telah dipilih akan memunculkan sebuah kotak dialog *GenericObjectEditor* yang memungkinkan *user* mengkonfigurasikan pilihan-pilihan pada filter. Untuk mengetahui informasi lebih lengkap tentang filter yang dipilih, *user* dapat meng-klik tombol *More.*

7) Tab Classify



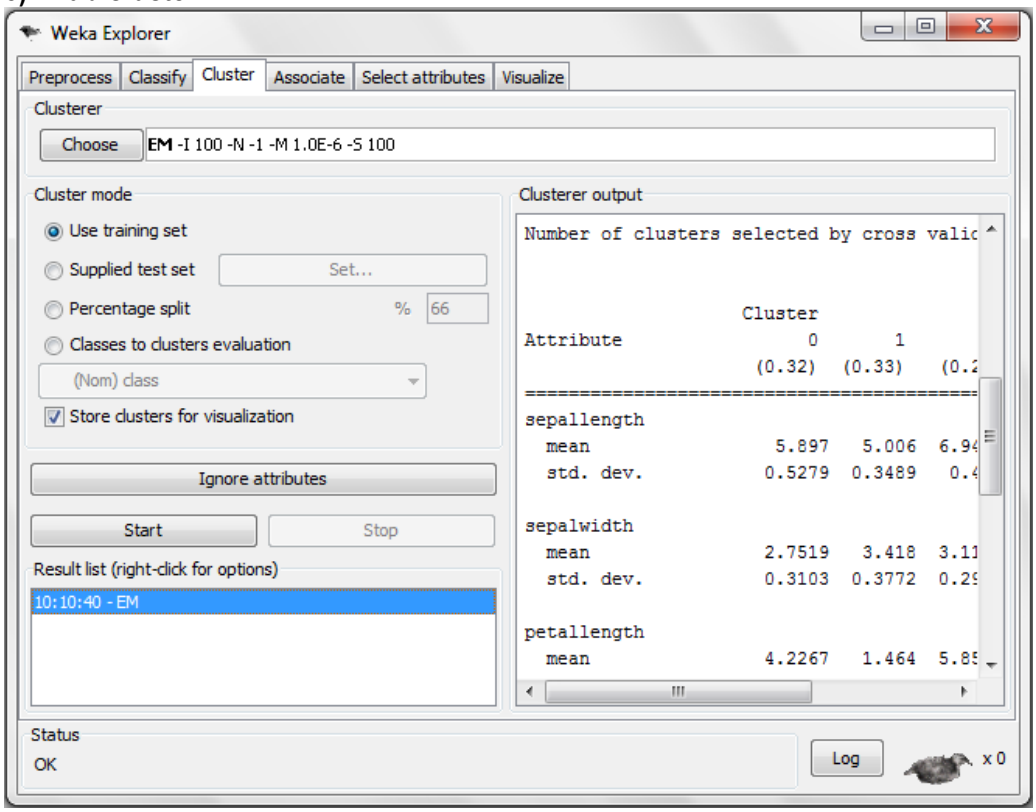
Tab ini memungkinkan *user* mengkonfigurasi dan mengeksekusi tiap *classifier* WEKA pada himpunan data tertentu. *User* dapat memilih *classifier* tertentu yang akan digunakan. Tersedia pula pilihan pengujian bagi *user* di bawah bagian pemilihan *classifiers,* yaitu:

- *Use training set. Classifier* dievaluasi pada kemampuannya memprediksi *class* dari *instances* yang diujikan.
- *Supplied test set.* Pengujian kemampuan *classifier* dilakukan terhadap himpunan *instances* terpisah yang di-*load* dari sebuah file.
- *Cross-validation*. *Classifier* dievaluasi dengan *cross-validation*, menurut jumlah *folds* yang dimasukkan pada kolom *Folds*.
- *Percentage split.* Evaluasi *clasifier* dilakukan pada sejumlah persentase tertentu dari data yang digunakan untuk pengujian.
- Setelah *classifier,* pilihan-pilihan pengujian, dan *class* telah ditentukan, proses pembelajaran dapat dimulai dengan klik tombol *Start*. *User* dapat menghentikan proses ini sewaktu-waktu dengan tombol *Stop*. Saat *training* selesai, area output *classifier* di sebelah kanan menampilkan teks yang menggambarkan hasil *training* dan pengujian. Sebuah entry baru juga muncul di kotak *Result list*.

Teks yang dihasilkan pada area output *classifier* berisi informasi tentang pilihan-pilihan skema, nama relasi, *instances*, atribut-atribut dan mode pengujian; model *classifier* dengan himpunan *training* lengkap, hasil mode pengujian yang dipilih, *summary*, akurasi terperinci menurut *class*, serta matriks *confusion.*

Errors klasifikasi dapat divisualisasikan dalam sebuah *tool* visualisasi data *pop-up*. Jika *classifier* menghasilkan sebuah *decision tree*, dapat ditampilkan secara grafis dalam sebuah *pop-up tree visualizer*.

8) Tab Cluster

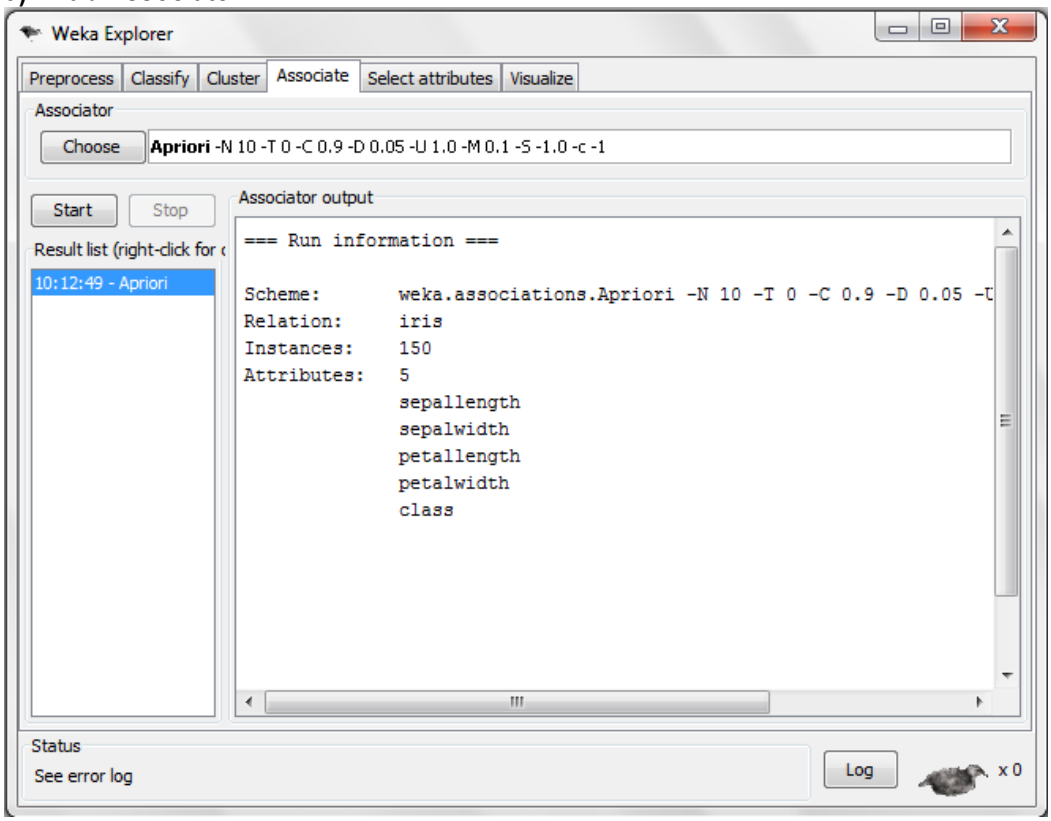


Tab ini serupa dengan *classification,* dengan sedikit perbedaan menurut *option* yang ditentukan *user.* Misalnya, *user* dapat dengan mudah mengabaikan beberapa atribut yang tidak diinginkan.

Dari tab ini *user* dapat mengkonfigurasi dan mengeksekusi tiap *clusterers* WEKA pada himpunan data tertentu untuk menemukan kelompok-kelompok dari *instances* yang sama dalam sebuah himpunan data. Skema-skema yang dapat diimplementasikan antara lain: *k*-Means, EM, Cobweb, *X*-means, FarthestFirst. *Clusters* dapat divisualisasikan dalam sebuah *tool* visualisasi data.

Kotak *cluster mode* digunakan untuk memilih apa yang akan di-*cluster* dan bagaimana melakukan evaluasi terhadap hasilnya. Tiga pilihan pertama serupa dengan yang terdapat pada klasifikasi: *Use training set, Supplied test set* dan *Percentage split* – kecuali bahwa sekarang data akan diolah dengan *clustering*. Mode keempat, *Classes to clusters evaluation*, membandingkan seberapa baik *clusters* yang terpilih sesuai dengan *class* yang telah ditentukan sebelumnya.

9) Tab Associate

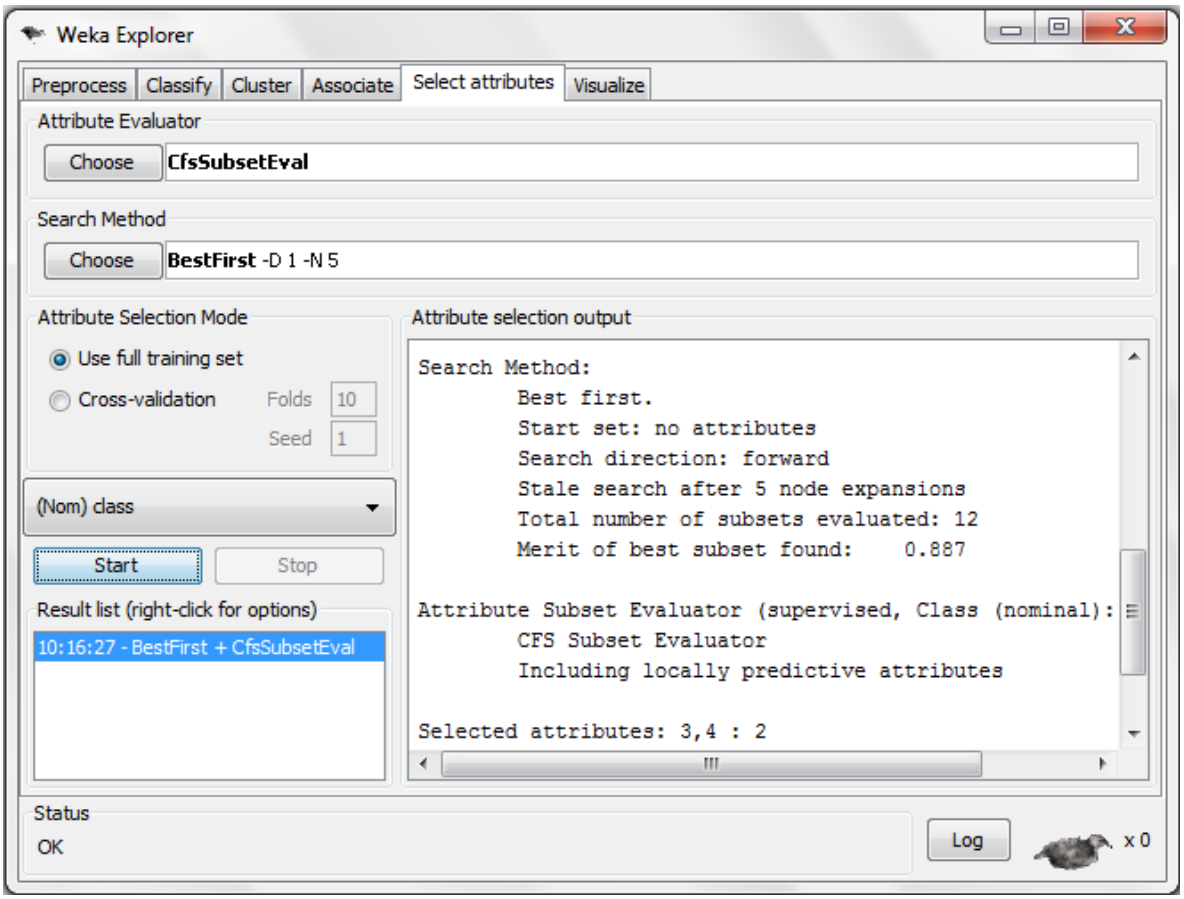


WEKA hanya mengimplementasikan sebuah algoritma untuk asosiasi, yaitu algoritma Apriori, untuk mempelajari aturan-aturan asosiasi. Asosiasi ini hanya bekerja dengan data diskrit untuk menentukan ketergantungan antara himpunan atribut. Apriori dapat menghitung seluluh aturan yang memenuhi nilai minimum *support* dan *confidence.*

Dari tab ini *user* dapat mempelajari himpunan data tertentu untuk menghasilkan aturan-aturan asosiasi menggunakan *associators* WEKA. Setelah parameter-parameter tertentu diset, klik tombol *Start.* Saat proses selesai dilakukan, klik kanan pada sebuah entry pada daftar hasil memungkinkan hasilnya dilihat atau disimpan.

10) Tab Select Atribute

WEKA juga menyediakan teknik-teknik untuk mengabaikan atribut-atribut yang tidak relevan dan/atau mengurangi dimensionalitas dari dataset. Setelah *loading* sebuah dataset, klik tab ini untuk memilih metode evaluasi (misalnya, *Principal Components Analysis, correlation-based*, *wrapper*, *information gain*, *chi-squared*,) dan metode pencarian (misalnya, *greedy, exhaustive, best-first, forward selection, random, genetic algorithm,* atau *ranking*). Berdasarkan kombinasi yang dipilih, waktu aktual yang diperlukan untuk pemilihan atribut dapat bervariasi atau sangat lama, bahkan untuk dataset kecil dengan sedikit atribut dan sedikit *instances*. Ingat bahwa tidak semua kombinasi metode evaluasi/pencarian valid, lihat pesan error di *Status bar.*



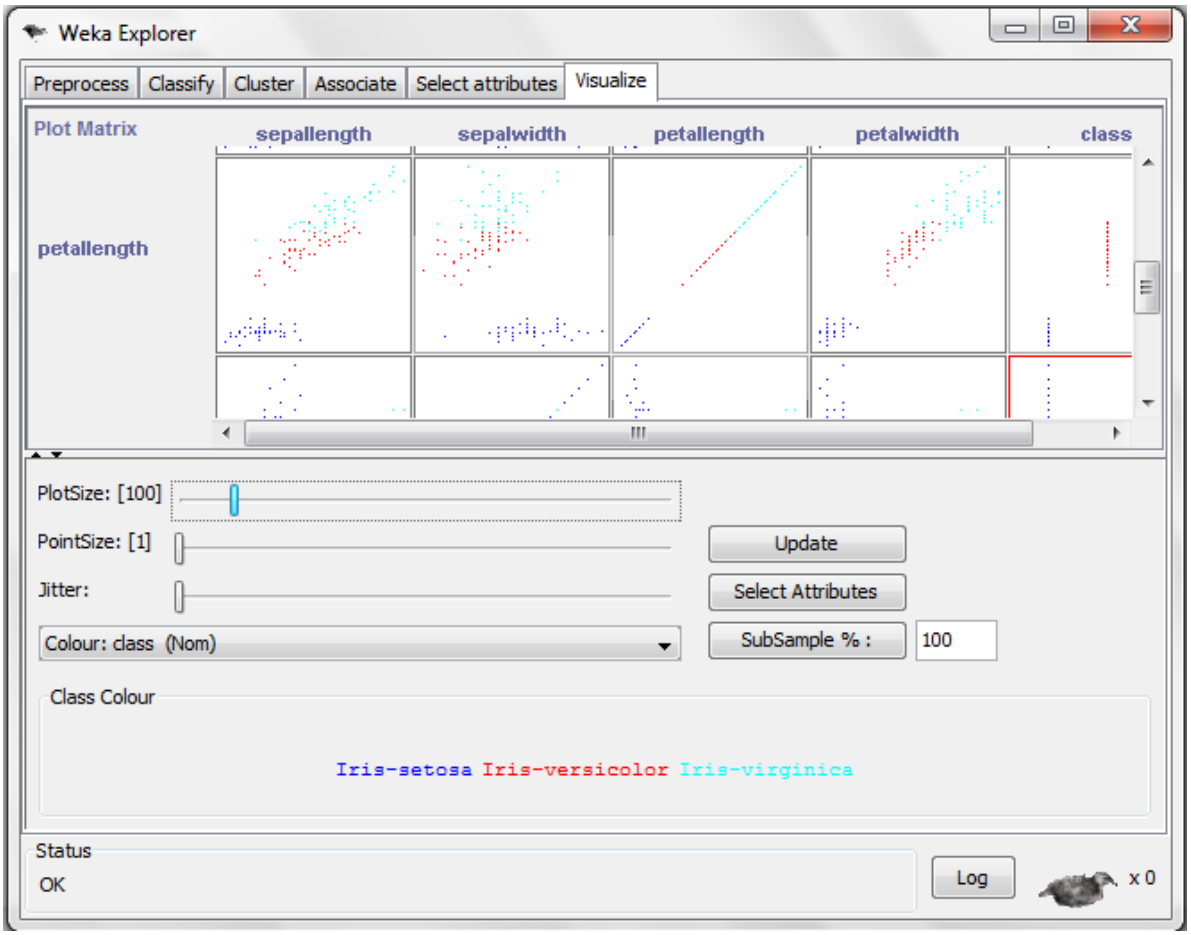
Pemilihan atribut terdiri atas pencarian pada seluruh kombinasi yang mungkin dari atribut-atribut pada data untuk menentukan subset atribut mana yang paling baik untuk prediksi. Untuk melakukannya, 2 objek harus diset: sebuah evaluator atribut dan sebuah metode pencarian. Evaluator menentukan metode yang digunakan untuk menilai tiap subset dari atribut. Metode pencarian menentukan gaya pencarian yang diterapkan.

Mode pemilihan atribut menyediakan 2 pilihan:

- *Use full training set*. Nilai subset atribut ditentukan dengan himpunan data *training* yang lengkap.
- *Cross-validation.* Nilai subset atribut ditentukan dengan sebuah proses validasi. *Fields* **Folds** dan **Seed** mengeset jumlah *folds* yang digunakan dan *seed* random yang digunakan saat melakukan *shuffle* pada data.

11) Tab Visualize

Tab ini menampilkan matriks plot 2 dimensi untuk himpunan data tertentu. Ukuran sel-sel individu dan titik-titik yang ditampilkan dapat dipilih dengan *slider* di bagian bawah tab. Jumlah sel dalam matriks dapat diubah dengan '*Select Attribute*' lalu memilih atribut tertentu untuk ditampilkan. Jika himpunan data besar, performansi *plotting* dapat ditingkatkan dengan menampilkan *subsample* himpunan data tertentu. Klik pada sebuah sel pada matriks menampilkan sebuah *window* tab plot yang lebih besar yang menampilkan *view* dari sel tersebut.



Tab ini juga dapat menampilkan *window* terpisah dari tab *classifier* dan tab *cluster* yang memungkinkan *user* memvisualisasikan prediksi yang dibuat oleh *classifiers/ clusterers*. Jika *class* diskrit, titik-titik yang diklasifikasikan dengan salah ditunjukkan dengan sebuah kotak berwarna sesuai *class* yang diprediksi oleh *classifier*; sedangkan jika *class* kontinu, ukuran tiap titik yang di-plot bervariasi dengan proporsi sesuai besarnya error yang dibuat oleh *classifier*.

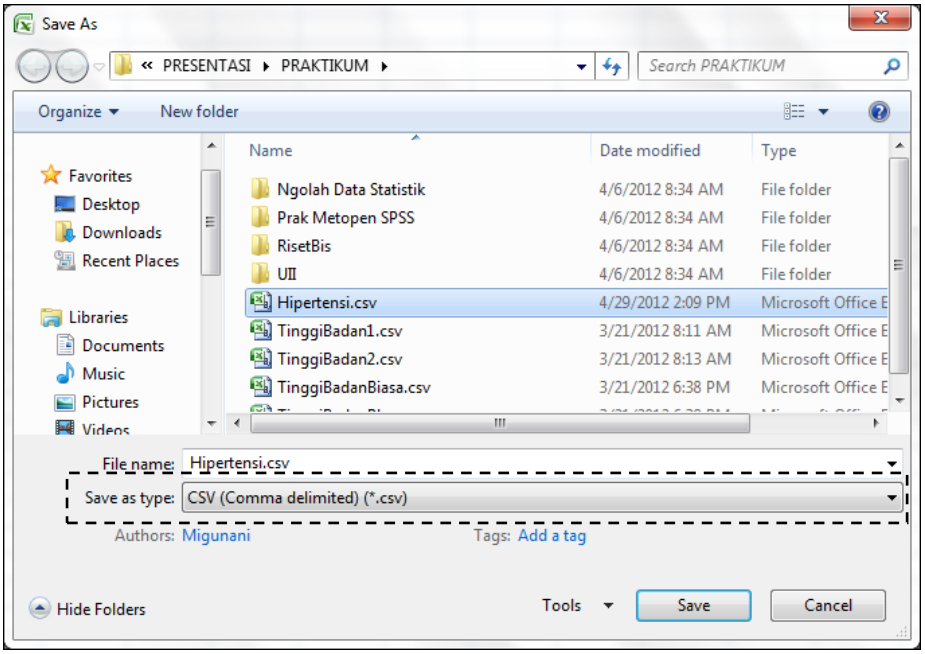
Titik-titik data diplot pada area utama pada *window*. Di bagian atas terdapat 2 tombol daftar *drop-down* untuk pemilihan sumbu x dan y yang diplot. *User* juga dapat memilih skema warna yang digunakan, berdasarkan atribut yang dipilih. Di bawahnya, sebuah *legend* mencatat nilai-nilai apa yang digambarkan oleh warna-warna tertentu. Jika nilainya diskrit, *user* dapat memodifikasi warna yang digunakan masing-masing dengan klik dan membuat sebuah seleksi yang sesuai pada *window* yang muncul.

1.6. Menggunakan perangkat lunak WEKA 3.6.0

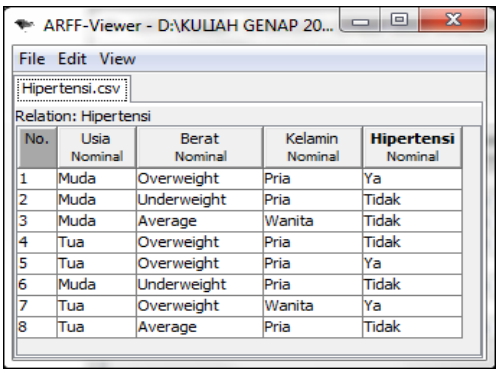
1) Data tabular pasien yang ada excel dikonversi terlebih dahulu kedalam format .csv, seting terlebih dahulu ***region and language*** (windows 7) dengan format english (*united state*).

Usia	Berat	Kelamin	Hipertensi
Muda	Overweight	Pria	Ya
Muda	Underweight	Pria	Tidak
Muda	Average	Wanita	Tidak
Tua	Overweight	Pria	Tidak
Tua	Overweight	Pria	Ya
Muda	Underweight	Pria	Tidak
Tua	Overweight	Wanita	Ya
Tua	Average	Pria	Tidak

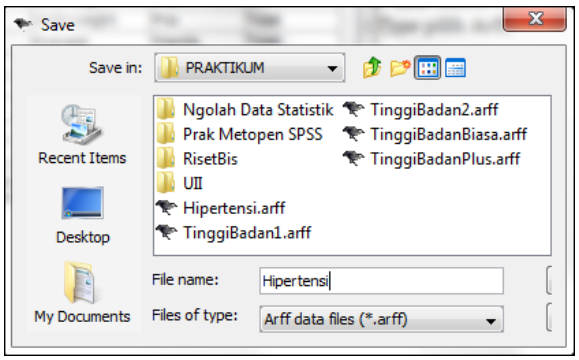
2) Simpan tabel diatas dengan Save As => pada bagian ***save as type*** pilih CSV (Comma Delimited), perhatikan gambar berikut :



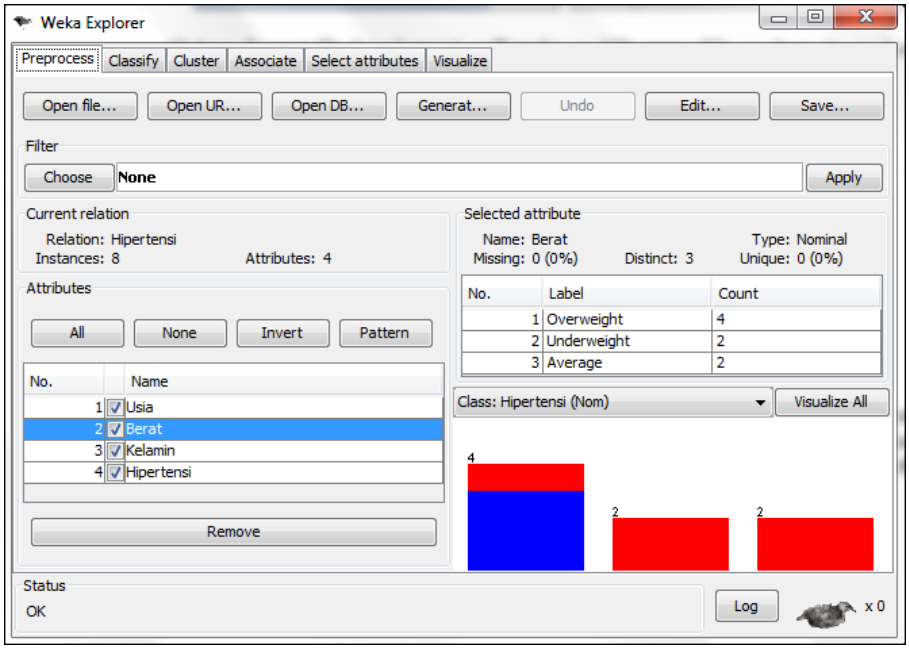
3) Buka aplikasi WEKA 3.6.0 >> klik menu Tools >> Arff Viewer >> Pilih file Hipertensi.csv



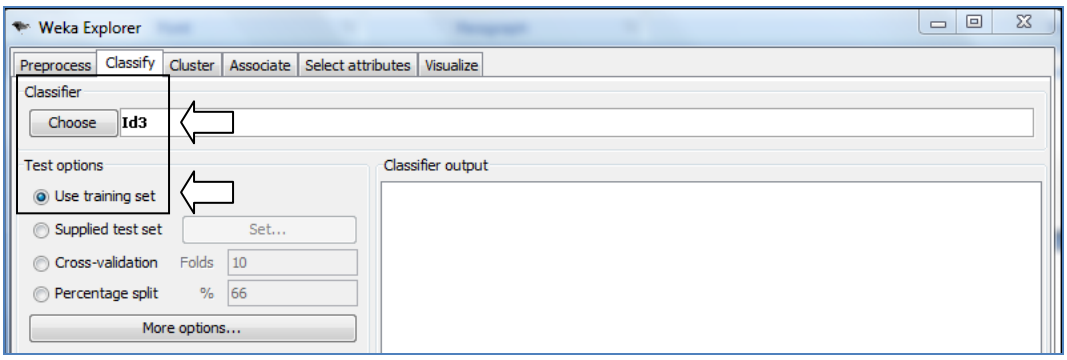
4) Agar format file berekstensi .arff maka >> klik menu File >> Save As >> beri nama file Hipertensi, pada pilihan *Save As Type* pilih Arff.



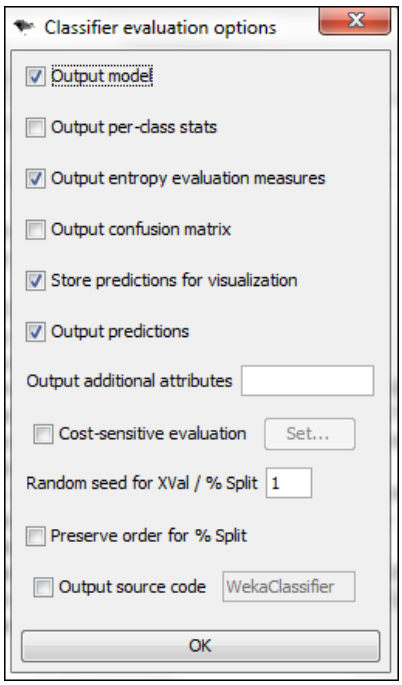
5) Setelah data training siap, selanjutnya akan diklasifikasi menggunakan mesin pembelajaran (*learning machine*) WEKA 3.6.0, pada jendela utama klik *Button Explorer* >> Pilih Tab Preprocces dan klik open file >> pilih file Hipertensi.arff



6) Klik Tab Classify >> Klik button Choose >> Pilih Tree >> Pilih ID3 >> pada options pilih >> Use Training Set seperti gambar berikut :



7) Pada opsi classifier centang *Output model*, *Output entropy evaluation measures, store prediction for visualization dan output prediction*.



1. *Output model* akan menghasilkan model pohon keputusan.
2. *Output entropy evaluation measures* akan menghasilkan hasil perhitungan *entropy.*
3. *Store prediction for visualization* akan menyimpan hasil prediksi klasifikasi untuk visualisasi pohon keputusan.
4. *Output predictor* akan menghasilkan prediksi hasil klasifikasi.

```
Scheme:       weka.classifiers.trees.Id3
Relation:     Hipertensi
Instances:    8
Attributes:   4
              Usia
              Berat
              Kelamin
              Hipertensi
Test mode:    evaluate on training data
```

8) Output hasil klasifikasi seperti gambar berikut.

```
Berat = Overweight
|   Usia = Muda: Ya
|   Usia = Tua
|   |   Kelamin = Pria: Ya
|   |   Kelamin = Wanita: Ya
Berat = Underweight: Tidak
Berat = Average: Tidak
```

```
inst#,     actual, predicted, error, probability
    1        1:Ya        1:Ya          *1      0
    2     2:Tidak     2:Tidak           0     *1
    3     2:Tidak     2:Tidak           0     *1
    4     2:Tidak        1:Ya      +  *0.5    0.5
    5        1:Ya        1:Ya        *0.5    0.5
    6     2:Tidak     2:Tidak           0     *1
    7        1:Ya        1:Ya          *1      0
    8     2:Tidak     2:Tidak           0     *1
```

Hasil klasifikasi menggunakan algoritma ID3 menunjukkan bahwa node awal adalah variabel Berat, selanjutnya variabel Usia dan Variabel jenis Kelamin.

BAB 5
FUNGSI ASSOSIASI DALAM DATAMINING

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan mechine learning WEKA 3.6.0 dan spreadsheet untuk mencari aturan assosiasi.
2. Mahasiswa dapat menjelaskan aturan assosiasi menggunakan algoritma *market basket analysis*.

1.2. Pendahuluan

Aturan asosiasi digunakan untuk menemukan hubungan antar item data yang ada. Aturan asosiasi diolah dari data-data dalam bentuk tabel yang terdiri dari kolom nomor transaksi dan jenis barang, yaitu a atribut (barang) {a_1, a_2, ..., an} dan k transaksi (instance). Pada aturan asosiasi akan dicari kombinasi yang paling sering terjadi dari suatu *itemset* dan menggambarkan hubungan kondisional serta hasil berupa aturan asosiasi. Algoritma assosiasi menggunakan data latihan untuk menghasilkan pengetahuan. Salah satu algoritma asosiasi adalah analisis keranjang belanja (*market basket analysis).* Pengetahuan yang diperoleh adalah berupa item-item belanja yang sering dibeli secara bersamaan dalam suatu waktu. Sehingga akan diperoleh aturan asosiasi dalam bentuk "if..then" atau "jika..maka" sebagai hasil dari analisis. Istilah-istilah umum dalam aturan asosiasi :

a. Item : barang yang dibeli/menjadi objek kegiatan belanja atau himpunan dari semua jenis item (dilambangkan dengan I).
b. Itemset : himpunan yang beranggotakan sebagian atau seluruh anggota I, misalnya I = {Asparagus, Beans, Broccoli, Corn, Green Peppers, Squash, Tomatoes}
c. Himpunan item yang dibeli oleh pengunjung ke-i adalah transaksi ke i yang dilambangkan dengan T_i.
   Misalnya T_1 = { Broccoli, Green Peppers, Corn }
   T_2 = { Asparagus, Squash, Corn }
d. Himpunan dari seluruh transaksi yang terjadi dilambangkan dengan D, misalnya D = { T_1, T_2, T_3..... T_n }
e. Aturan assosiasi yang akan dihasilkan akan berbentuk implikasi, "Jika A, maka B" atau "A=>B", A disebut anteseden (pendahulu) implikasi, sedangkan B disebut konsekuen (pengikut) implikasi.
f. Aturan asososiasi yang akan dihasilkan haruslah memenuhi dua sifat, yaitu : baik A maupun B adalah himpunan bagian murni dari I dan himpunan A dan B adalah dua himpunan yang saling lepas.
g. Ukuran kinerja pada aturan assosiasi
   - Support (ukuran tingkat dominasi suatu itemset dengan keseluruhan transaksi)

$$s(A => B) = P(A \cup B) = \frac{\text{Jml transaksi yg mengandung item } A \cup B}{\text{Jumlah Total Transaksi Pada D}}$$

   - Confidence (ukuran yang menunjukkan hubungan kondisional antar dua itemset)

$$conf(A => B) = P(A \mid B) = \frac{\text{Jml transaksi yg mengandung item } A \cup B}{\text{Jumlah Transaksi Item A}}$$

1.3. Menghitung nilai support dan confidence dan menghasilkan aturan asosiasi menggunakan excel.

Buatlah terlebih dahulu data transaksi belanja berikut ini menggunakan microsoft excel.

Daftar Transaksi Belanja

Transaksi	Item Belanja
1	Asparagus, Beans, Broccoli, Corn
2	Asparagus, Corn
3	Beans, Corn
4	Corn, Beans, Broccoli
5	Beans, Corn
6	Asparagus, Beans
7	Asparagus, Broccoli, Corn

Cel B3

Himpunan Item (I) = 7 Itemset

1	Asparagus (As)
2	Beans (Be)
3	Broccoli (Br)
4	Corn (Co)

Counting Itemset

	Transaksi	As	Be	Br	Co
1	1	1	1	1	1
1	2	1	0	0	1
1	3	0	1	0	1
1	4	0	1	1	1
1	5	0	1	0	1
1	6	1	1	0	0
1	7	1	0	1	1
	SUM	4	5	3	6

Cel E3

Cel F3

Cel E9

Cel E3 = IFERROR(IF(FIND("Asparagus",B3)>0,1,0),0)

Cel F3 = IFERROR(IF(FIND("Beans",B3)>0,1,0),0)

Cel G3 = IFERROR(IF(FIND("Broccoli",B3)>0,1,0),0)

Cel H3 = IFERROR(IF(FIND("Corn",B3)>0,1,0),0)

Ditetapkan Nilai Minimum Support 20%, Minimum Confidence 60%

Untuk menghitung prosesntase *support* dan *confidence* dari semua kemungkinan asosiasi berdasarkan transaksi yang terjadi, buatlah terlebih dahulu tabel kemungkinan berikut :

No	X			Y			n(X U Y)	N	% Support	n(X)	% Confidence	Apakah Rule ?
J5							Q5		Q7	Q8	Q9	Q10
1	As			Be			2	7	29%	4	50%	x
2	As			Br			2	7	29%	4	50%	x
3	As			Co			3	7	43%	4	75%	✓
4	As			Be	Br		1	7	14%	4	25%	x
5	As			Br	Co		2	7	29%	4	50%	x
6	As			Be	Co		1	7	14%	4	25%	x
7	As			Be	Br	Co	1	7	14%	4	25%	x
8	Be			As			2	7	29%	5	40%	x
9	Be			Br			2	7	29%	5	40%	x
10	Be			Co			4	7	57%	5	80%	✓
11	Be			As	Br		1	7	14%	5	20%	x
12	Be			Br	Co		2	7	29%	5	40%	x
13	Be			As	Co		1	7	14%	5	20%	x
14	Be			As	Br	Co	1	7	14%	5	20%	x
15	Br			As			2	7	29%	3	67%	✓
16	Br			Be			2	7	29%	3	67%	✓
17	Br			Co			3	7	43%	3	100%	✓
18	Br			As	Be		1	7	14%	3	33%	x
19	Br			Be	Co		2	7	29%	3	67%	✓
20	Br			As	Co		2	7	29%	3	67%	✓
21	Br			As	Be	Co	1	7	14%	3	33%	x
22	Co			As			3	7	43%	6	50%	x
23	Co			Be			4	7	57%	6	67%	✓
24	Co			Br			3	7	43%	6	50%	x
25	Co			As	Be		1	7	14%	6	17%	x
26	Co			Be	Br		2	7	29%	6	33%	x
27	Co			As	Br		2	7	29%	6	33%	x
28	Co			As	Be	Br	1	7	14%	6	17%	x
29	As	Be		Br			1	7	14%	9	11%	x
30	As	Be		Co			1	7	14%	9	11%	x
31	As	Be		Br	Co		1	7	14%	9	11%	x
32	As	Br		Be			1	7	14%	7	14%	x
33	As	Br		Co			2	7	29%	7	29%	x
34	As	Br		Be	Co		1	7	14%	7	14%	x
35	As	Co		Be			1	7	14%	10	10%	x
36	As	Co		Br			2	7	29%	10	20%	x
37	As	Co		Be	Br		1	7	14%	10	10%	x
38	Be	Br		As			1	7	14%	8	13%	x
39	Be	Br		Co			2	7	29%	8	25%	x
40	Be	Br		As	Cp		1	7	14%	8	13%	x

41	Be	Co		As			1	7	14%	11	9%	x
42	Be	Co		Br			2	7	29%	11	18%	x
43	Be	Co		As	Br		1	7	14%	11	9%	x
44	Br	Co		As			2	7	29%	9	22%	x
45	Br	Co		Be			2	7	29%	9	22%	x
46	Br	Co		As	Be		1	7	14%	9	11%	x
47	As	Be	Br	Co			1	7	14%	12	8%	x
48	As	Be	Co	Br			1	7	14%	15	7%	x
49	As	Br	Co	Be			1	7	14%	13	8%	x
50	Be	Br	Co	As			1	7	14%	14	7%	x

n(X U Y) =

=SUMIFS(Bernilai_Satu,IF(K6="As",As_,IF(K6="Be",Be_,IF(K6="Br",Br_,IF(K6="Co",Co_,Bernilai_Satu)))),1,
IF(L6="As",As_,IF(L6="Be",Be_,IF(L6="Br",Br_,IF(L6="Co",Co_,Bernilai_Satu)))),1,IF(M6="As",As_,IF(M6="Be",Be_,IF(M6
="Br",Br_,IF(M6="Co",Co_,Bernilai_Satu)))),1,IF(N6="As",As_,IF(N6="Be",Be_,IF(N6="Br",Br_,IF(N6="Co",Co_,Bernilai_Sa
tu)))),1,IF(O6="As",As_,IF(O6="Be",Be_,IF(O6="Br",Br_,IF(O6="Co",Co_,Bernilai_Satu)))),1)

% Support = +Q6/R6

n(X)=
IFERROR(HLOOKUP(K6,Bind_Record,ROWS(Bind_Record),FALSE),0)+IFERROR(HLOOKUP(L6,Bind_Recor
d,ROWS(Bind_Record),FALSE),0)+IFERROR(HLOOKUP(M6,Bind_Record,ROWS(Bind_Record),FALSE),0)

% Confidence = +Q6/T6

Is In Rule = IF(AND(S6>=Min_Support,U6>=Min_Confidence),1,0)

Sehingga diperoleh Rule sebagai berikut :

No	X			Y			n(X U Y)	N	% Support	n(X)	% Confidence	Apakah Rule ?
3	As			Co			3	7	43%	4	75%	✓
10	Be			Co			4	7	57%	5	80%	✓
15	Br			As			2	7	29%	3	67%	✓
16	Br			Be			2	7	29%	3	67%	✓
17	Br			Co			3	7	43%	3	100%	✓
19	Br			Be	Co		2	7	29%	3	67%	✓
20	Br			As	Co		2	7	29%	3	67%	✓
23	Co			Be			4	7	57%	6	67%	✓

1.4. Menghitung nilai support dan confidence dan menghasilkan aturan asosiasi algoritma apriori menggunakan WEKA.

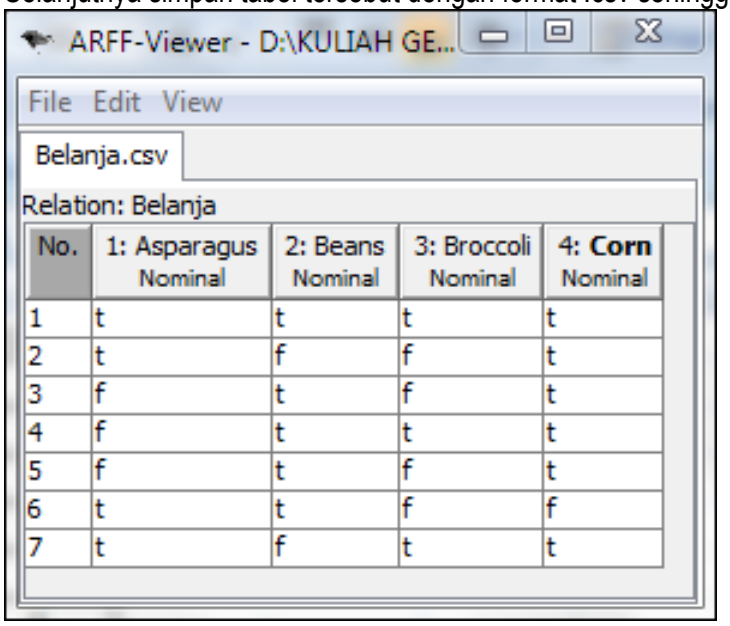
Apriori mencari aturan asosiasi berdasarkan parameter assosiasi yaitu nilai support dan *confidence* yang memenuhi *minimal support* dan *minimal confidence*. Aturan asosiasi diperoleh dengan melihat nilai *confidence* yang dihasilkan.

1. Dengan menggunakan excel rubahlah terlebih dahulu data transaksi sebelumnya seperti tabel berikut :

Asparagus	Beans	Broccoli	Corn
t	t	t	t

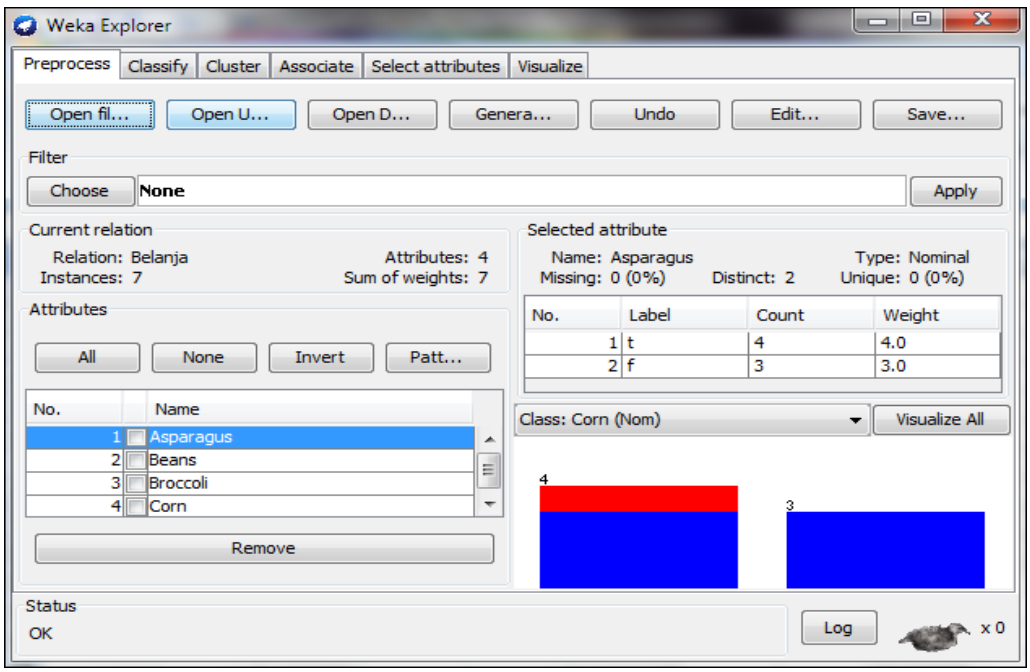
t	f	f	t
f	t	f	t
f	t	t	t
f	t	f	t
t	t	f	f
t	f	t	t

2. Selanjutnya simpan tabel tersebut dengan format .csv sehingga dapat dibaca oleh WEKA.



No.	1: Asparagus Nominal	2: Beans Nominal	3: Broccoli Nominal	4: Corn Nominal
1	t	t	t	t
2	t	f	f	t
3	f	t	f	t
4	f	t	t	t
5	f	t	f	t
6	t	t	f	f
7	t	f	t	t

3. Aktifkan Eksplorer WEKA dan buka file .csv yang akan di olah menggunakan aturan assosiasi dengan algoritma apriori.



4. Untuk mengolah data menggunakan algoritma apriori, klik tab associate >> Klik Buton Choose >> Klik Start untuk mengolah data dan melihat hasilnya.

```
=== Run information ===

Scheme:       weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation:     Belanja
Instances:    7
Attributes:   4
              Asparagus
              Beans
              Broccoli
              Corn
=== Associator model (full training set) ===

Apriori
=======

Minimum support: 0.35 (2 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 13

Generated sets of large itemsets:

Size of set of large itemsets L(1): 7
Size of set of large itemsets L(2): 14
Size of set of large itemsets L(3): 7
Size of set of large itemsets L(4): 1

Best rules found:

 1. Asparagus=f 3 ==> Beans=t 3    <conf:(1)> lift:(1.4) lev:(0.12) [0] conv:(0.86)
 2. Asparagus=f 3 ==> Corn=t 3    <conf:(1)> lift:(1.17) lev:(0.06) [0] conv:(0.43)
 3. Broccoli=t 3 ==> Corn=t 3    <conf:(1)> lift:(1.17) lev:(0.06) [0] conv:(0.43)
 4. Asparagus=f Corn=t 3 ==> Beans=t 3    <conf:(1)> lift:(1.4) lev:(0.12) [0] conv:(0.86)
 5. Asparagus=f Beans=t 3 ==> Corn=t 3    <conf:(1)> lift:(1.17) lev:(0.06) [0] conv:(0.43)
 6. Asparagus=f 3 ==> Beans=t Corn=t 3    <conf:(1)> lift:(1.75) lev:(0.18) [1] conv:(1.29)
 7. Beans=f 2 ==> Asparagus=t 2    <conf:(1)> lift:(1.75) lev:(0.12) [0] conv:(0.86)
 8. Beans=f 2 ==> Corn=t 2    <conf:(1)> lift:(1.17) lev:(0.04) [0] conv:(0.29)
 9. Beans=f Corn=t 2 ==> Asparagus=t 2    <conf:(1)> lift:(1.75) lev:(0.12) [0] conv:(0.86)
10. Asparagus=t Beans=f 2 ==> Corn=t 2    <conf:(1)> lift:(1.17) lev:(0.04) [0] conv:(0.29)
```

BAB 6
THEOREMA BAYES

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan mechine learning WEKA 3.6.0 dan spreadsheet untuk klasifikasi dan prediksi menggunakan theorema bayes.
2. Mahasiswa dapat menjelaskan teknik klasifikasi dan prediksi kelas menggunakan theorema bayes.

1.2. Pendahuluan

Klasifikasi berdasarkan theorema bayes adalah pendekatan statistik yang fundamental dalam pengenalan pola (*pattern recognition*). Pendekatan ini berdasarkan pada kuantitas *trade-off* antara berbagai keputusan klasifikasi dengan menggunakan probabilitas dan biaya yang ditimbulkan (apabila terjadi kesalahan klasifikasi). Misalnya terdapat masalah yang bersifat hipotesis yakni mendesain sebuah fungsi klasifikasi untuk memisahkan dua jenis objek misalnya ikan bandeng dan ikan kakap.

Bila h1 mewakili ikan bandeng, h2 mewakili ikan kakap, maka kemunculan keduanya bersifat probabilistik. Jika jumlah masing-masing ikan bandeng dan ikan kakap sama, maka peluang munculnya ikan tersebut juga akan sama besar. Maka probabilitas prior p(h1) dan p(h2) masing-masing menyatakan peluang kemunculan ikan bandeng dan kakap. Probablilitas prior ini menyatakan perkiraan kita akan jenis ikan yang akan muncul sebelum benar-benar muncul melalui konveyor.

Misalnya N adalah jumlah total ikan yang ditangkap, N1 jumlah ikan bandeng dan N2 jumlah ikan kakap, maka estimasi munculnya ikan bandeng adalah **p(h1) = N1/N** dan ikan kakap adalah **p(h2) = N2/N**. Namun pada banyak kasus kita akan mengambil keputusan dengan informasi tambahan yang lebih banyak, tidak sekedar menggunakan probabilitas prior saja. Misalnya informasi tambahan untuk kasus diatas adalah **variabel warna (x)** untuk meningkatkan keakuratan prediksi. Jenis ikan yang berbeda akan menghasilkan pembacaan warna yang berbeda. Dengan mempertimbangkan warna (x) sebagai variabel random kontinyu yang distribusinya tergantung ikan yang muncul dan dinyatakan dengan p(x\h) artinya peluang muncul x jika diketahui h.

Sehingga p(x\h1) dan p(x\h2) menyatakan perbedaan distribusi ikan bandeng dan ikan kakap. Kita sudah mengetahui probabilitas prior : $p(h_j)$ dan probabilitas bersayarat $p(x|h_j)$, dimana j=1,2, sehingga fungsi peluang (*likelihood*) dari h_j terhadap x adalah $P(x\backslash h_j)$. Dari tambahan informasi berupa *likelihood* $P(x\backslash h_j)$ dan warna x, kita peroleh probabilitas posterior. Maka probabilitas posterior / probabilitas munculnya h_j Jika diketahui x (teorema bayes) dirumuskan :

$$P(h_j \mid x) = \frac{p(x \mid h_j)P(h_j)}{p(x)}$$

Maka evidence dalam kasus kategori dua kelas tersebut (bandeng & kakap) adalah :

$$P(x) = \sum_{i=1}^{2} p(x \mid h_j)P(h_j)$$

Posterior = likelihood x prior
evidence

Hasil penjumlahan seluruh probabilitas posterior adalah satu. Maka aturan bayes ditetapkan jika P(h1|x) < P(h2|x) maka x diklasifikasikan sebagai h2. Sementara Probabilitas error pada bayes adalah :

P(error | x) = P(h1 | x), jika kita putuskan h2
P(h2 | x), jika kita putuskan h1

Sehingga kita dapat meminimalkan error jika diberikan nilai x dengan memutuskan :

1) h1 jika P(h1|x) > P(h2|x)
2) h2 jika P(h1|x) < P(h2|x)

1.3. Bayes Learning

Misalnya kita memiliki beberapa alternatif hipotesis h bagian dari H. Dalam bayes learning kita ingin memaksimalkan hipotesis yang paling mungkin (minimum apriori / MAP), jika diberi data x, maka dapat dirumuskan :

$$h_{MAP} = \arg\max P(h \mid x)$$

$$h_{MAP} = \arg\max \frac{P(x \mid h)P(h)}{P(x)}$$

$$h_{MAP} = \arg\max P(x \mid h)P(h)$$

Persamaan diatas bisa disederhanakan, dengan mempertimbangkan nilai p(x|h) untuk menentukan hipotesis yang paling mungkin. Sehingga hipotesis maksimum likelihood dinotasikan sebagai berikut :

h_{ML}= arg max $_{hEH}$ P(x | h)

Dalam konteks datamining data x adalah himpunan training dan H adalah ruang data.

1.4. Studi Kasus Sederhana

Terdapat dua hipotesis yaitu (1) pasien pengidap kanker, (2) pasien tidak mengidap kanker. Data tersedia dari uji laboratorium dengan dua kemungkinan yaitu (+) atau (-). Informasi prior menunjukkan bahwa dari keseluruhan populasi hanya 0,008 menderita kanker. Uji lab menghasikan output (+) sebesar 98% dari semua kasus pengidap kanker, dan output (-) sebesar 97% dimana pasien tidak mengidap kanker. Dari kasus diatas dapat dirangkum :

- P(kanker) = 0.008, P(~kanker) = 0.992
- P(+\kanker) = 0.98, P(-\kanker) = 0.02
- P(+\~kanker) = 0.03, P(-\~kanker) = 0.97

Misalnya ada uji lab baru dan hasilnya (+), apa kesimpulan kita akan pasien tersebut ?. Maka kita bisa menghitung semua kemungkinan dan memilih hipotesis maksimum apriori sebagai berikut :

- P(kanker\+) = P(+\kanker) P(kanker) = (0.98)(0.008) = 0.0078
- P(~kanker\+) = P(+\~kanker)P(~kanker) = (0.03)(0.992) = 0.0298

Sehingga HMAP - ~kanker, probabilitas posterior dihitung secara lengkap adalah :

$$\mathrm{P}(\text{kanker} \mid +) = \frac{0.0078}{0.0078 + 0.0298} = 0.21$$

1.5. Studi Kasus Klasifier Bayes

Terdapat 14 data dengan output bermain sepak bola atau tidak. Setiap data ditandai dengan atribut cuaca, temperatur, kelembaban dan angin. Kita akan menggunakan naive bayes untuk menentukan kelas dari data berikut :

- *Cuaca = cerah, Temperatur = dingin, Kelembaban = tinggi, dan Angin = besar*

Data tersebut belum ada di dalam tabel, namun akan diklasifikasikan dengan diprediksi (data no. 15) menggunakan klasifier bayes berikut :

Data	Cuaca	Temp	Kelembaban	Angin	Main
	x1	x2	x3	x4	y
1	Cerah	Panas	Tinggi	Kecil	**Tidak**
2	Cerah	Panas	Tinggi	Besar	**Tidak**
3	Mendung	Panas	Tinggi	Kecil	**Ya**
4	Hujan	Sedang	Tinggi	Kecil	**Ya**
5	Hujan	Dingin	Normal	Kecil	**Ya**
6	Hujan	Dingin	Normal	Besar	**Tidak**
7	Mendung	Dingin	Normal	Besar	**Ya**
8	Cerah	Sedang	Tinggi	Kecil	**Tidak**
9	Cerah	Dingin	Normal	Kecil	**Ya**
10	Hujan	Sedang	Normal	Kecil	**Ya**
11	Cerah	Sedang	Normal	Besar	**Ya**
12	Mendung	Sedang	Tinggi	Besar	**Ya**
13	Mendung	Panas	Normal	Kecil	**Ya**
14	Hujan	Sedang	Tinggi	Besar	**Tidak**
15	**Cerah**	**Dingin**	**Tinggi**	**Besar**	**?**

Hitung probabilitas masing-masing variabel, misalnya :

- P(main) = 9/14 = 0.64
- P(tidak) = 5/14 = 0.36
- P(Angin = besar | main) = 3/9 = 0.33
- P(Angin = besar | tidak) = 3/5 = 0.60
- P(main) P(cerah\main) P(dingin\main) P(tinggi\main) P(besar\main)
  = 9/14 * 2/9 * 3/9 * 3/9 * 3/9 = **0.0053**
- P(tidak) P(cerah\tidak) P(dingin\tidak) P(tinggi\tidak) P(besar\tidak)
  = 5/14 * 3/5 * 1/5 * 4/5 * 3/5 = **0.0206** (**lebih besar**)

Sehingga dengan naive bayes kita simpulkan untuk tidak main dengan data input ini. Karena nilai probabilitas "tidak main" lebih besar dibandingkan nilai probabilitas "ya". Dengan normalisasi agar probabilitas sama dengan 1, kita bisa menghitung probabilitas tidak main jika diberikan nilai-nilai atribut, untuk contoh ini probabilitasnya :

$$\frac{P(Tidak\_Main)}{P(Tidak\_Main)+P(Main)} = \frac{0.0206}{0.0206+0.0053} = 0.795$$

1.6. Teorema bayes menggunakan menggunakan microsoft excel untuk prediksi

1. Buatlah tabel keputusan bermain sepak bola berikut :

Data	Cuaca	Temp	Kelembaban	Angin	Main
	x1	x2	x3	x4	y
1	Cerah	Panas	Tinggi	Kecil	**Tidak**
2	Cerah	Panas	Tinggi	Besar	**Tidak**
3	Mendung	Panas	Tinggi	Kecil	**Ya**
4	Hujan	Sedang	Tinggi	Kecil	**Ya**
5	Hujan	Dingin	Normal	Kecil	**Ya**
6	Hujan	Dingin	Normal	Besar	**Tidak**
7	Mendung	Dingin	Normal	Besar	**Ya**
8	Cerah	Sedang	Tinggi	Kecil	**Tidak**
9	Cerah	Dingin	Normal	Kecil	**Ya**
10	Hujan	Sedang	Normal	Kecil	**Ya**
11	Cerah	Sedang	Normal	Besar	**Ya**
12	Mendung	Sedang	Tinggi	Besar	**Ya**
13	Mendung	Panas	Normal	Kecil	**Ya**
14	Hujan	Sedang	Tinggi	Besar	**Tidak**
15	**Cerah**	**Dingin**	**Tinggi**	**Besar**	**?**

Cel A4

Cel F17

Nilai Probabilitasya :

1. **P(Main) =(COUNTIF(F4:F17,"Ya"))/(COUNTIF(F4:F17,"Ya")+(COUNTIF(F4:F17,"Tidak")))**
2. **P(Tidak) =(COUNTIF(F4:F17,"Tidak"))/(COUNTIF(F4:F17,"Ya")+(COUNTIF(F4:F17,"Tidak")))**
3. **P(Angin=Besar|Main) =3/9**
4. **P(Angin=Besar|Tidak) =3/5**
5. **P(main) x P(cerah\main) x P(dingin\main) x P(tinggi\main) x P(besar\main)**
   = 9/14 * 2/9 * 3/9 * 3/9 * 3/9
6. **P(tidak) P(cerah\tidak) P(dingin\tidak) P(tinggi\tidak) P(besar\tidak)**
   = 5/14 * 3/5 * 1/5 * 4/5 * 3/5

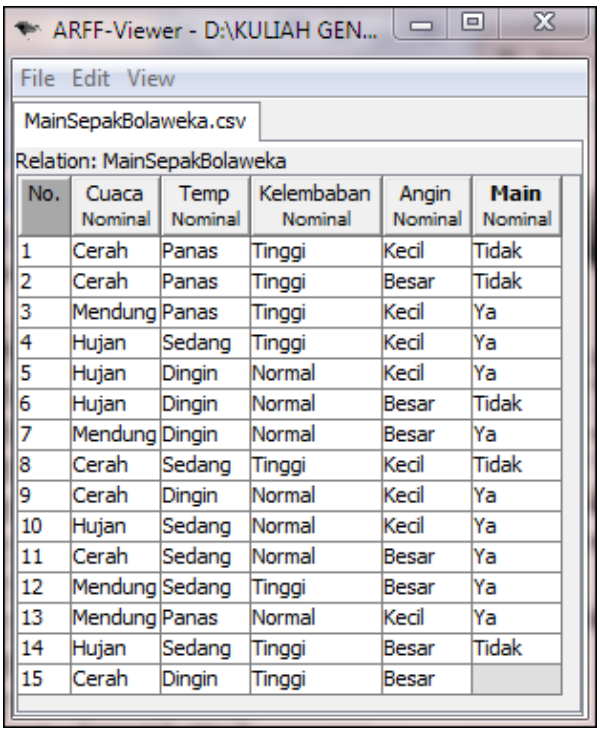
Dari hasil perhitungan Nilai probabilitas "Tidak Main" lebih besar dari Nilai probabilitas "Main" >> P(Tidak Main) > P(Main)

Hasil Normalisasi

$$\frac{P(Tidak\_Main)}{P(Tidak\_Main)+P(Main)} = \frac{0.0206}{0.0206+0.0053} = 0.795 \text{ (Tidak Main Bola)}$$

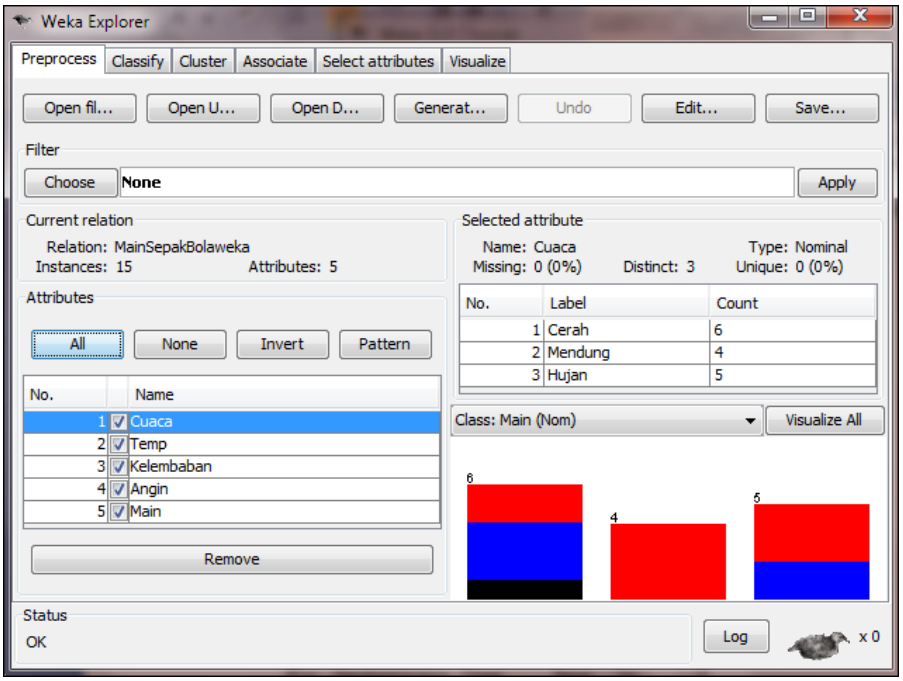
1.7. Teorema bayes menggunakan menggunakan Weka untuk prediksi.

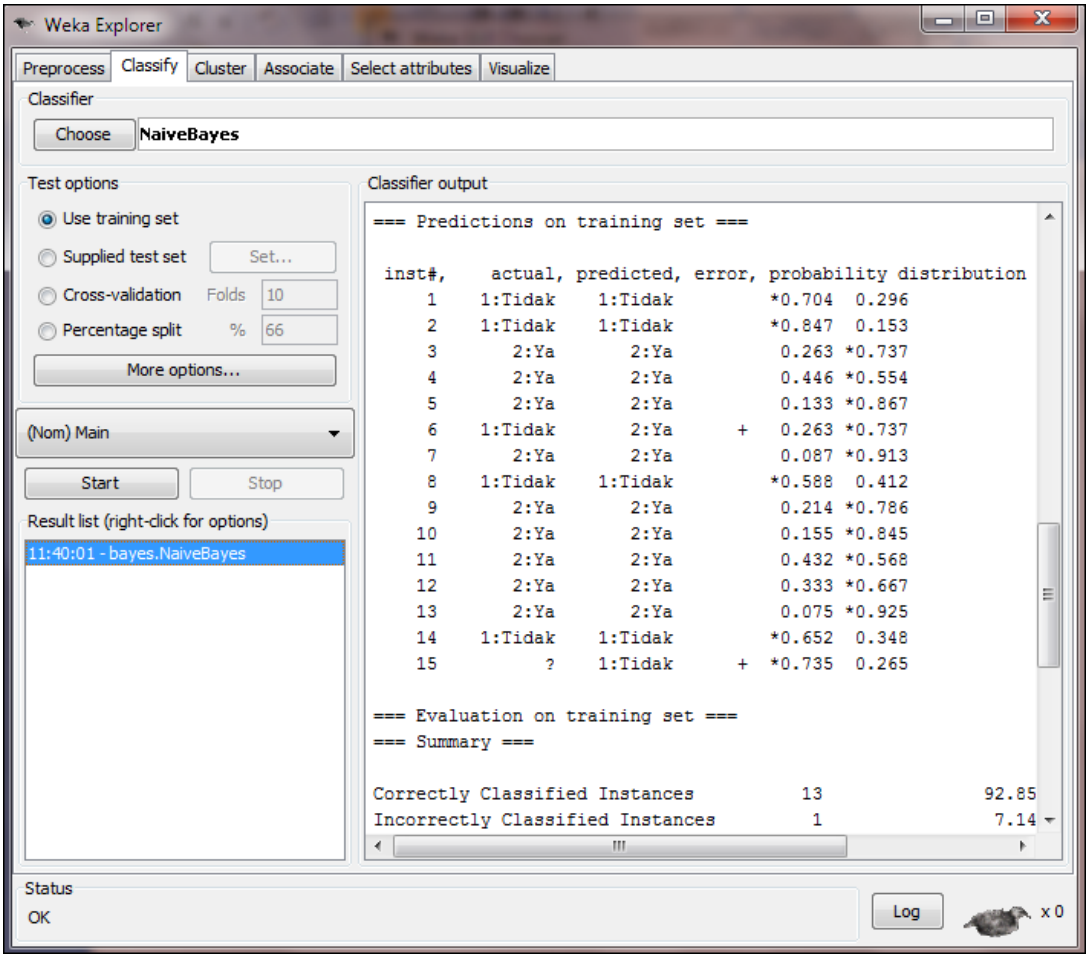
1. Konversikan terlebih dahulu data excel tersebut ke file .csv agar bisa dibaca menggunakan *Weka Machine Learning*.
2. Buka file hasil konversi menggunakan arff viewer di Weka *Machine Learning* dan simpan dengan format .arff



No.	Cuaca Nominal	Temp Nominal	Kelembaban Nominal	Angin Nominal	**Main** Nominal
1	Cerah	Panas	Tinggi	Kecil	Tidak
2	Cerah	Panas	Tinggi	Besar	Tidak
3	Mendung	Panas	Tinggi	Kecil	Ya
4	Hujan	Sedang	Tinggi	Kecil	Ya
5	Hujan	Dingin	Normal	Kecil	Ya
6	Hujan	Dingin	Normal	Besar	Tidak
7	Mendung	Dingin	Normal	Besar	Ya
8	Cerah	Sedang	Tinggi	Kecil	Tidak
9	Cerah	Dingin	Normal	Kecil	Ya
10	Hujan	Sedang	Normal	Kecil	Ya
11	Cerah	Sedang	Normal	Besar	Ya
12	Mendung	Sedang	Tinggi	Besar	Ya
13	Mendung	Panas	Normal	Kecil	Ya
14	Hujan	Sedang	Tinggi	Besar	Tidak
15	Cerah	Dingin	Tinggi	Besar	

3. Buka data pada eksplorer Weka, kemudian pilih tab Classify >> pilih Bayes >> NaiveBayes >> Klik More Options >> Centang Output Predictor >> Ok





=== Predictions on training set ===

inst#,	actual,	predicted,	error,	probability distribution
1	1:Tidak	1:Tidak		*0.704 0.296
2	1:Tidak	1:Tidak		*0.847 0.153
3	2:Ya	2:Ya		0.263 *0.737
4	2:Ya	2:Ya		0.446 *0.554
5	2:Ya	2:Ya		0.133 *0.867
6	1:Tidak	2:Ya	+	0.263 *0.737
7	2:Ya	2:Ya		0.087 *0.913
8	1:Tidak	1:Tidak		*0.588 0.412
9	2:Ya	2:Ya		0.214 *0.786
10	2:Ya	2:Ya		0.155 *0.845
11	2:Ya	2:Ya		0.432 *0.568
12	2:Ya	2:Ya		0.333 *0.667
13	2:Ya	2:Ya		0.075 *0.925
14	1:Tidak	1:Tidak		*0.652 0.348
15	**?**	**1:Tidak**	**+**	***0.735 0.265**

4. Hasil klasifikasi dan predikasi untuk data ke-15 menghasilkan “tidak” dengan nilai probabilitas 0.735 dan distribusi sebesar 0.265

BAB 7
FUNGSI PENGELOMPOKKAN (CLUSTERING)

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan mechine learning WEKA 3.6.0 untuk pengelompokkan dan prediksi menggunakan algorithma k-means.
2. Mahasiswa dapat menjelaskan teknik pengelompokkan (*clustering*) mengguna kan algorithma k-means.

1.2. Pendahuluan

Pada pengelompokkan, hal-hal yang akan dikelompokkan disebut dengan objek atau catatan. Setiap objek dibedakan dengan objek lainya berdasarkan atribut yang dimiliki masing-masing. Kumpulan dari seluruh atribut disebut dengan data input. Pengetahuan yang hendak diperoleh pada fungsi ini adalah berupa penentuan beberapa kelompok objek/catatan yang memiliki kemiripan atribut. Pada pengelompokkan, objek-objek yang memiliki kemiripan atribut akan dikelompokkan kedalam salah satu dari sekian kelompok. Adapun objek-objek yang kurang memiliki kesamaan atribut akan ditempatkan pada kelompok yang berbeda.

Tujuan dari pengelompokan sekumpulan data obyek ke dalam beberapa kelompok yang mempunyai karakteristik tertentu dan dapat dibedakan satu sama lainnya adalah untuk proses analisis dan interpretasi lebih lanjut sesuai dengan tujuan penelitian yang dilakukan. Analisis cluster dapat diterapkan pada bidang apa saja. Namun pemakaian teknik ini lebih familiar pada bidang pemasaran karena memang salah satu kegiatan yang dilakukan dalam pemasaran adalah pengelompokan, yang disebut segmentasi pasar.

Konsep dasar pengukuran analisis cluster adalah konsep pengukuran jarak (*distance*) dan kesamaan (*similarity*). Distance adalah ukuran tentang jarak pisah antar obyek sedangkan similarity adalah ukuran kedekatan Konsep ini penting karena pengelompokan pada analisis cluster didasarkan pada kedekatan. Pengukuran jarak (*distance type measure*) digunakan untuk data-data yang bersifat matriks, sedangkan pengukuran kesesuaian (matching type measure) digunakan untuk data-data yang bersifat kualitatif. Banyak metode dikembangkan dari konsep jarak. Untuk mengukur jarak dua titik x dan y (d(x,y)) kita dapat menggunakan konsep jarak dengan syarat-syarat :

1)d(x,y) ≥ 0 (non-negatif)
Tidak ada jarak yang memiliki nilai negatif
2)d(x,y) = 0 jika dan hanya jika x=y (identity of indescernibles)
Jarak antara satu objek atau titik dengan objek atau titik itu sendiri adalah nol
3)d(x,y) = d(y,x) (simetri)
Jarak dari x ke y adalah sama dengan jarak dari y ke x
4)d(x,z) ≤ d(x,y) + d(y,z) (ketidak samaan segitiga)

Konsep jarak dalam literatur machine learning diantaranya jarak euclidian, manhattan atau cityblok, minkowski, chebeshev dan mahalanobis.

- Euclidean Distance merupakan ukuran jarak antara dua item X dan Y dengan persamaan
- Squared Euclidean $d(x,y) = \sqrt{\sum (x_i, y_i)^2}$
- Chebychev $d(x,y) = \max_i | x_i, y_i |$ $d(x,y) = \sum (x_i, y_i)^2$

- Manhattan / City Block $D(X,Y) = \sum | x_i, y_i |$
- Minkowski $d(x,y) = \left[\sum | x_i, y_i |^2 \right]^{1/p}$

Dari beberapa teknik pengelompokkan yang paling umum digunakan adalah klastering k-means. Dalam teknik ini data akan dikelompokkan dalam k kelompok atau klaster. Untuk mengelompokkan data nilai k harus ditentukan terlebih dahulu. Biasanya user telah mempunyai informasi awal tentang objek yang sedang dipelajari, termasuk berapa jumlah klaster yang paling tepat. Secara detail kita dapat menggunakan ukuran ketidak miripan dalam mengelompookan objek. Ketidak miripan bisa diterjemahkan dalam konsep jarak. Jika jarak antara dua objek cukup dekat, maka dapat dikatakan dua objek itu mirip. Semakin dekat berarti semakin tinggi kemiripanya. Semakin tinggi nilai jarak maka akan semakin tinggi ketidakkemiripanya.

Algoritma k-means klastering memiliki tahapan dalam pengelompokanya sebagai berikut :

1. Pilih atau tetapkan jumlah kelompok atau klaster (k)
2. Inisiasi k pusat klaster ini bisa dilakukan dengan berbagai cara, yang paling sering dilakukan adalah dengan cara random, sehingga nilai pusat-pusat klaster diberi nilai awal dengan angka random.
3. Tempatkan klaster objek ke klaster terdekat berdasarkan jarak kedua objek tersebut. Demikian juga kedekatan anatara data degan pusat kluster terntentu ditetantukan berdasarkan jarak anatara data dengan pusat klaster. Pada tahap ini perlu dihitung jarang semua data pada pusat klaster k, sehingga jarak terdekat anatara data dengan pusat klaster akan menentukan data termasuk kelompok atau klaster mana.
4. Hitung kembali pusat klaster dengan keanggotaan klaster yang sekarang. Pusat klaster adalah rata-rata dari semua data dalam klaster tertentu. Jika dikehendaki dapat juga menggunakan nilai median dari klaster tersebut.
5. Tugaskan lagi setiap objek dengan memakai pusat klaster yang baru. Jika pusat klaster tidak berubah lagi, maka proses pengklasteran selesai.

Berikut ini contoh data yang akan dikelompokkan menggunakan algoritma k-means. Terdapat 4 data / objek dan masing-masing memiliki 2 attribut sebagai berikut :

Objek	Attr X	Attr Y
A	1	1
B	2	1
C	4	3
D	5	4

1. Langkah awal adalah menetapkan jumlah klaster/kelompok (k) adalah 2 (dua), kemudian ditetapkan pusat data (*centroid*) dari data tersebut misalnya data ke-1 dan data ke-2 yaitu c1=(1,1) dan c2= (2,1).
2. Langkah berikutnya menghitung jarak data dengan pusat data c1 dan c2, yaitu :

   Jarak Objek A=(1,1) dengan c1=(1,1) adalah $= \sqrt{(0-0)^2 + (0-0)^2} = 0$

   Jarak Objek A=(1,1) dengan c2=(2,1) adalah $= \sqrt{(1-2)^2 + (1-1)^2} = 1$

   Jarak Objek B=(2,1) dengan c1=(1,1) adalah $= \sqrt{(2-1)^2 + (1-1)^2} = 1$

   Jarak Objek B=(2,1) dengan c2=(2,1) adalah $= \sqrt{(2-2)^2 + (1-1)^2} = 0$

Jarak Objek C=(4,3) dengan c1=(1,1) adalah $=\sqrt{(4-1)^2+(3-1)^2}=3{,}61$

Jarak Objek C=(4,3) dengan c2=(2,1) adalah $=\sqrt{(4-2)^2+(3-1)^2}=2{,}83$

Jarak Objek D=(5,4) dengan c1=(1,1) adalah $=\sqrt{(5-1)^2+(4-1)^2}=5{,}00$

Jarak Objek D=(5,4) dengan c2=(2,1) adalah $=\sqrt{(5-2)^2+(4-1)^2}=4{,}24$

3. Hasil perhitungan jarak ini disimpan dalam bentuk matriks k x n, dengan k banyaknya cluster dan n banyak obyek. Setiap kolom dalam matriks tersebut menunjukkan obyek sedangkan baris pertama menunjukkan jarak ke centroid pertama, baris kedua menunjukkan jarak ke centroid kedua. Matriks jarak setelah iterasi ke-0 adalah sebagai berikut :

$$\begin{matrix} A & B & C & D & \\ [1 & 2 & 4 & 5] & X \\ [1 & 1 & 3 & 4] & Y \end{matrix}$$

$$Do=\begin{matrix}[0 & 1 & 3{,}61 & 5{,}00 & c1=(1,1) & \text{cluster-1} \\ [1 & 0 & 2{,}83 & 4{,}24 & c2=(2,1) & \text{cluster-2}\end{matrix}$$

Dengan memasukkan setiap obyek ke dalam cluster (grup) berdasarkan jarak minimumnya. Jadi objek A dimasukkan ke grup 1, dan objek B, C dan D memasukkan ke grup 2. Keanggotaan obyek ke dalam grup dinyatakan dengan matrik, elemen dari matriks bernilai 1 jika sebuah obyek menjadi anggota grup.

$$G^0=\begin{matrix}\begin{bmatrix}1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1\end{bmatrix} & \begin{matrix}\text{cluster-1} \\ \text{cluster-2}\end{matrix} \\ A\,B\,C\,D & \end{matrix}$$

4. Iterasi-1, menetukan centroid baru berdasarkan anggota masing-masing grup, selanjutnya ditentukan centroid baru. Grup 1 hanya berisi 1 obyek, sehingga centroidnya tetap c1=(1,1). Grup 2 mempunyai 3 anggota, sehingga centroidnya ditentukan berdasarkan rata-rata koordinat ketiga anggota tersebut

$$c2=\left(\frac{2+4+5}{3},\frac{1+3+4}{3}\right)=(3.66,2.66)$$

5. Iterasi-1, menghitung jarak obyek ke centroid, selanjutnya menghitung jarak antara centroid baru dengan seluruh obyek dalam grup dihitung kembali sehingga diperoleh matriks jarak sebagai berikut :

$$\begin{matrix} A & B & C & D & \\ [1 & 2 & 4 & 5] & X \\ [1 & 1 & 3 & 4] & Y \end{matrix}$$

$$Do = \begin{matrix}[0 & 1 & 3{,}61 & 5{,}00 & c1 = (1,1) & \text{cluster-1} \\ [3{,}14 & 2{,}36 & 0{,}47 & 1{,}89 & c2 = (3.66,2.66) & \text{cluster-2}\end{matrix}$$

6. Iterasi-1, clustering obyek: langkah ke-3 diulang kembali, menentukan keanggotaan grup berdasarkan jaraknya. Berdasarkan matriks jarak yang baru, maka objek B harus dipindah ke grup 2.

$$G^1 = \begin{matrix}\begin{bmatrix}1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1\end{bmatrix} & \begin{matrix}\text{cluster-1} \\ \text{cluster-2}\end{matrix} \\ A\ B\ C\ D & \end{matrix}$$

7. Iterasi-2, menentukan centroid: langkah ke-4 diulang kembali untuk menentukan centroid baru berdasarkan keanggotaan grup yang baru. Grup 1 dan grup 2 masing-masing mempunyai 2 anggota, sehingga centroidnya menjadi :

$$c1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = (1.5,1) \qquad c2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = (4.5,3.5)$$

8. Iterasi-2, menghitung jarak obyek ke centroid : ulangi langkah ke-2, sehingga diperoleh matriks jarak sebagai berikut:

$$\begin{matrix} A & B & C & D & \\ [1 & 2 & 4 & 5\,] & X \\ [1 & 1 & 3 & 4\,] & Y \end{matrix}$$

$$Do = \begin{matrix}[0.5 & 0.5 & 3{,}20 & 4{,}61 & c1 = (1.5,1) & \text{cluster-1} \\ [4{,}30 & 3{,}54 & 0{,}71 & 0{,}71 & c2 = (4.5,3.5) & \text{cluster-2}\end{matrix}$$

9. Iterasi-2, clustering obyek mengelompokkan tiap-tiap obyek berdasarkan jarak minimumnya, diperoleh:

$$G^2 = \begin{matrix}\begin{bmatrix}1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1\end{bmatrix} & \begin{matrix}\text{cluster-1} \\ \text{cluster-2}\end{matrix} \\ A\ B\ C\ D & \end{matrix}$$

10. Hasil pengelompokkan pada iterasi terakhir dibandingkan dengan hasil sebelumnya, diperoleh $G^2 = G^1$. Hasil ini menunjukkan bahwa tidak ada lagi obyek yang berpindah grup, dan algoritma telah stabil. Hasil akhir clustering ditunjukkan dalam Tabel 1.

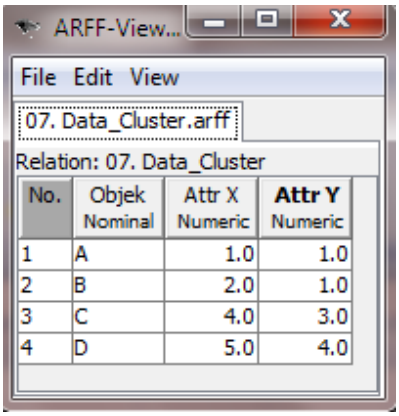
Objek	Attr X	Attr Y	Cluster
A	1	1	1
B	2	1	1
C	4	3	2
D	5	4	2

1.3. Algoritma K-mean menggunakan weka 3.60 untuk pengelompokan (*clustering*).

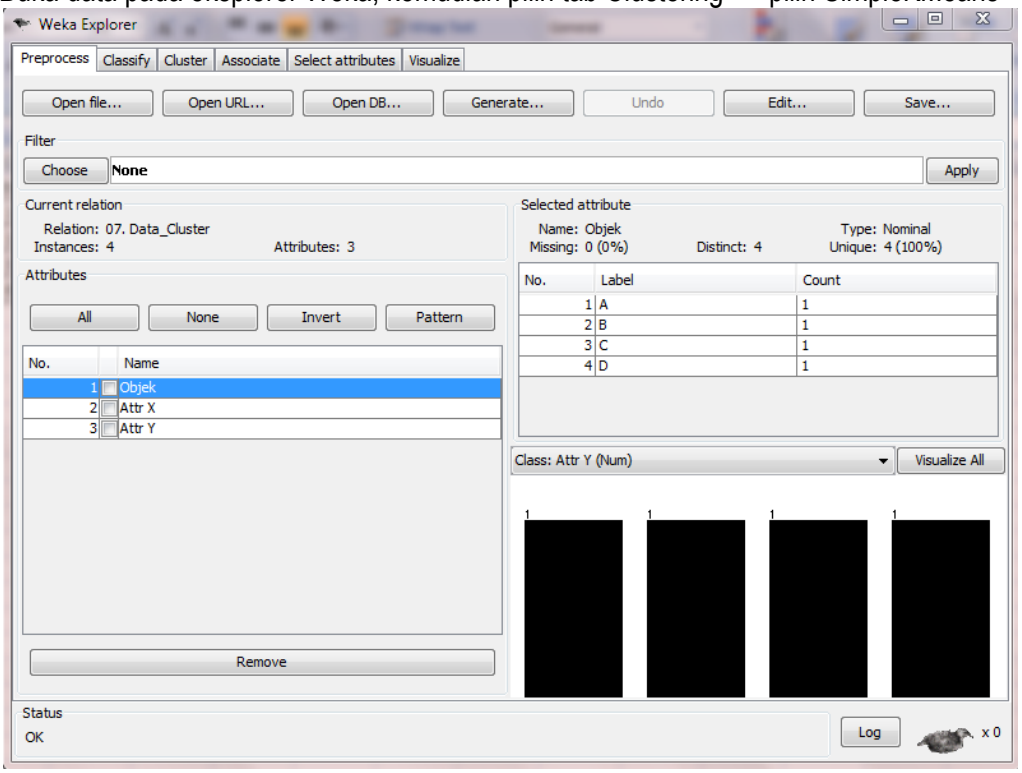
1. Konversikan terlebih dahulu data excel ke file .csv agar bisa dibaca menggunakan *Weka Machine Learning*.

Objek	Attr X	Attr Y
A	1	1
B	2	1
C	4	3
D	5	4

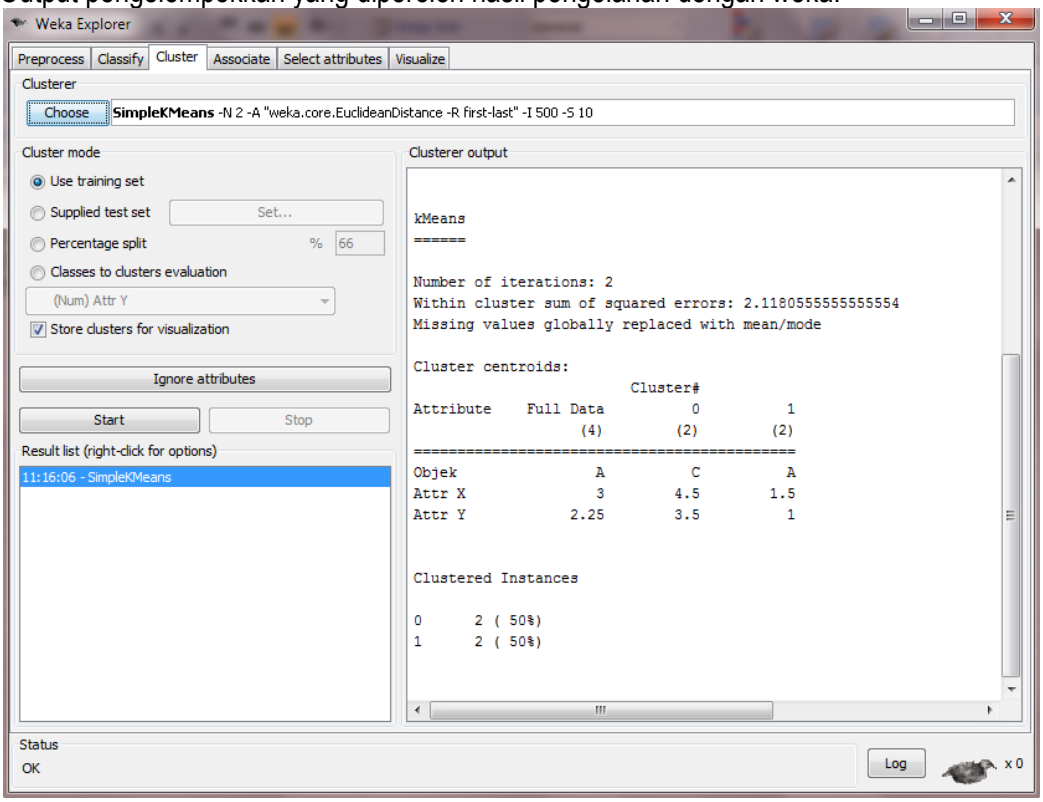
2. Buka file hasil konversi menggunakan arff viewer di Weka *Machine Learning* dan simpan dengan format .arff



5. Buka data pada eksplorer Weka, kemudian pilih tab Clustering >> pilih SimpleKMeans >> Ok



6. Output pengelompokkan yang diperoleh hasil pengolahan dengan weka.



=== Run information ===

Scheme: weka.clusterers.SimpleKMeans -N 2 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10
Relation: 07. Data_Cluster
Instances: 4
Attributes: 3
Objek
Attr X
Attr Y
Test mode: evaluate on training data

=== Model and evaluation on training set ===
kMeans
======
Number of iterations: 2
Within cluster sum of squared errors: 2.1180555555555554
Missing values globally replaced with mean/mode

Cluster centroids:
Cluster#
Attribute Full Data 0 1
(4) (2) (2)
Objek A C A
Attr X 3 4.5 1.5
Attr Y 2.25 3.5 1

Clustered Instances
0 2 (50%)
1 2 (50%)

1.4. Algoritma K-mean menggunakan excel untuk pengelompokan umur dan berat badan.

1)Buatlah tabel data umur dan berat badan mahasiswa menggunakan excel sebagai berikut

Data Umur dan Berat Badan

Data	Umur	Berat
Objek 1	25	50
Objek 2	26	48
Objek 3	35	65
Objek 4	67	60
Objek 5	55	65
Objek 6	53	60
Objek 7	34	48
Objek 8	48	50

Cel A4
Cel B6
Cel B9
Cel C11

2)Tentukan Centroid ke-1 (pusat data) sebagai cluster awal dari data yang ada misalnya objek ke-3 sebagai cluster ke-1 dan objek ke-7 sebagai cluster ke-2 sebagai berikut :

Centroid ke-1

Cluster 1	35	65
Cluster 2	34	48

Cel F3
Cel G4

3) Hitung jarak data dengan centroid ke-1 untuk masing-masing data dengan dengan formula

$$d(x,y) = \sqrt{(y1-x1)^2 + (y2-x2)^2}$$

Jarak Data Dengan Centroid ke-1

Instance	Jarak Ke Cluster 1	Jarak Ke Cluster 2	Cluster
1	18.02775638	9.219544457	2
2	19.23538406	8	2
3	0	17.02938637	1
4	32.38826948	35.11409973	1
5	20	27.01851217	1
6	18.68154169	22.47220505	1
7	17.02938637	0	2
8	19.84943324	14.14213562	2

Cel J4
Cel K4
Cel L4
Cel L11

- Formula Cel J4 = SQRT(((F3-B4)^2+(G3-C4)^2)), untuk Cel J5 sampai J11 dapat mengkopi rumus dari Cel J4.
- Formula Cel K4 = SQRT(((F4-B4)^2+(G4-C4)^2)), untuk Cel K5 sampai K11 dapat mengkopi rumus dari Cel K4.
- Formula Cel L4 = IF(J4<K4,1,2), untuk Cel L5 sampai L11 dapat mengkopi rumus dari Cel L4.

4) Setelah Cluster 1 dan Cluster 2 dapat dipisahkan seperti tabel diatas, selanjutnya kita hitung kembali Centroid ke-2 dengan menghitung rata-rata (average) yang termasuk dalam cluster 1 dan cluster 2.

Iterasi Ke-1

Tentukan Centroid Baru

Cluster Baru Diperoleh

Cluster 1	52.5	62.5
Cluster 2	33.25	49

Cel F14

Cel G15

Kelompok data yang termasuk Cluster-1 (Cel F14 dan F15) adalah objek ke 3,4,5 dan 6 maka dihitung rata-rata umur dan berat sebagai berikut :

a) Formula Cluster1(X,Y) dimana X = umur, maka X = Average(X3,X4,X5,X6) maka formula Cel F14=Average(B6:B9), B6:B9 lihat cel pada **tabel data umur dan berat badan** diatas.
b) Formula Cluster1(X,Y) dimana Y = berat, maka Y = Average(Y3,Y4,Y5,Y6) atau formula Cel G15=Average(C6:C9), C6:C9 lihat cel pada **tabel data umur dan berat badan** diatas.

Kelompok data yang termasuk Cluster-2 adalah objek ke 3,4,5 dan 6 maka dihitung rata-rata umur dan berat sebagai berikut :

a) Formula Cluster1(X,Y) dimana X = umur, maka X = Average(X1,X2,X7,X8) maka formula Cel F15=Average (B4:B5,B10:B11), B4:B5,B10:B11 lihat cel pada **tabel data umur dan berat badan** diatas.
b) Formula Cluster1(X,Y) dimana Y = berat, maka Y = Average(Y1,Y2,Y7,Y8) atau formula Cel G15=Average (C4:C5,C10:C11), C6:C9 lihat cel pada **tabel data umur dan berat badan** diatas.

5) Hitung jarak data dengan centroid ke-2 untuk masing-masing data dan cari kelompok Cluster seperti pada langkah No.3 sehingga menghasilkan tabel berikut ini.

Jarak Data Dengan Centroid

Instance	Jarak Ke Cluster 1	Jarak Ke Cluster 2	Cluster
1	30.20761493	8.310385069	2
2	30.20761493	7.318640584	2
3	17.67766953	16.0954186	2
4	14.71393897	35.49735906	1
5	3.535533906	27.00115738	1
6	2.549509757	22.60669149	1
7	23.50531855	1.25	2
8	13.28533026	14.78385944	1

6) Lakukan langkah yang sama seperti langkah 3, 4 dan 5 sampai tidak terjadi perubahan Cluster.

BAB 8
ARTIFICIAL NEURAL NETWORK (ANN)

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan mechine learning neuroxl untuk peramalan (forecasting) dan estimasi menggunakan algorithma.
2. Mahasiswa dapat menjelaskan teknik pengelompokkan (*clustering*) mengguna kan algorithma k-means.

1.2. Pendahuluan

Artificial Neural Network (ANN) awalnya mendapat inspirasi dari jaringan syaraf makluk hidup. ANN terinspirasi dari jaringan yang sangat komplek yang terdiri dari neuron yang saling terhubung. ANN menawarkan kelebihanya dimana bisa mengatasi persoalan tanpa mengadakan perubahan drastis terhadap modelnya. Ada beberapa karakteristik kemampuan otak manusia, dimana diharapkan ANN dapat meniru cara kerja otak manusia diantaranya mengingat, menghitung, mengeneralisasi, adaptasi dan konsumsi energi yang rendah.

Komputer digital mampu mengalahkan otak manusia dalam hal menghitung angka. Tetapi manusia mampu melakukan kerja yang sangat susah dilakukan komputer seperti mengenali orang secara cepat dilingkungan yang ramai hampir tanpa usaha keras. ANN berusaha meniru struktur dan cara kerja otak manusia sehingga mampu menggantikan pekerjaan manusia. Pekerjaan mengenali pola, prediksi, klasifikasi, pendekatan fungsi dan optimasi. Kelebihan ANN diantaranya mampu menyelesaiakan pekerjaan prediksi yang polanya nonlinier, waktu penyelesaian yang cepat, robust terhadap missing data. Kegunaan ANN diantaranya peramalan curah hujan, pendeteksian tornado, pendeteksian pemalsuan pemakaian kartu kredit dan lain-lain.

Pada 1940, McCulloch dan Pitt melakukan riset tentang ANN, kemudian Pada 1960, Rosenblatt menemukan teknik perceptron, kemudian pada periode sesudah 1960, Minsky dan Papaert membuktikan kelemahan perceptron sederhana yang ditemukan Rosenblatt. McCulloch-Pitt mengajukan unit batas binari sebagai model komputasi untuk ANN. Model ini menghitung jumlah dari n signal input $x_{i,j}$ = 1,2,...,n yang diberi bobot dan menghasilkan nilai 1 (satu) bila jumlah tersebut diatas batas tertentu dan O (nol) bila dibawah batas tersebut. Secara matematis persamaan bisa ditulis (Haykin, 1999).

$$y = \varphi\left(\sum_{j}^{n} w_j x_j - u\right)$$

Dimana φ= fungsi aktivasi, x = signal input, dan w = bobot input ke-j

Sebuah neuron adalah unit pemrosesan informasi yang sangat vital dalam operasi NN Elemen-lemen dasar model neuron adalah

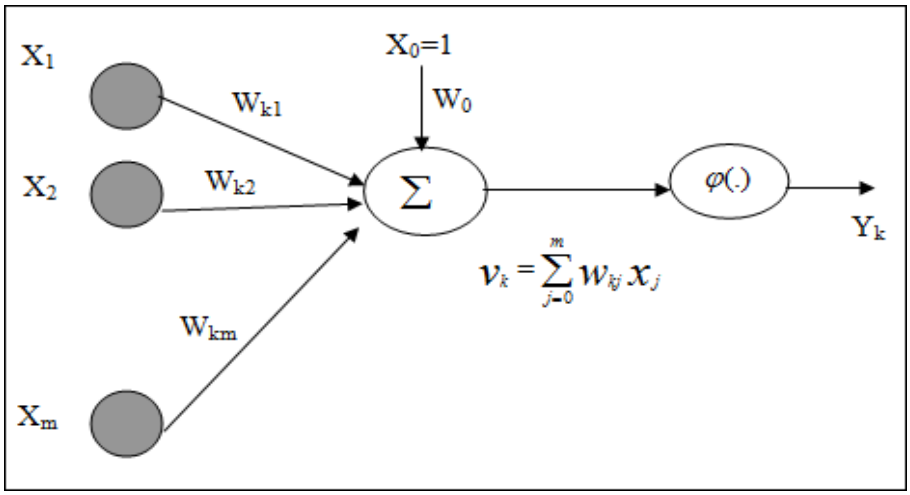
a) Set sinapsis atau link penghubung, yang ditandai dengan adanya bobot atau kekuatan dari link ini.

b) Penambah, untuk menjumlahkan signal input yang diberi bobot.

c) Fungsi aktivasi, untuk membatasi jumlah output dari suatu neuron.

Dalam model neuron memiliki nilai bias dinyatakan dalam bk atau wo,bias bk mempunyai fungsi untuk menaikkan dan menurunkan net input untuk fungsi aktivasi, tergantung nilainya, positif atau negatif. Neuran k bisa didiskripsikan secara matematis.

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \qquad yk = \varphi(u_k + bk)$$



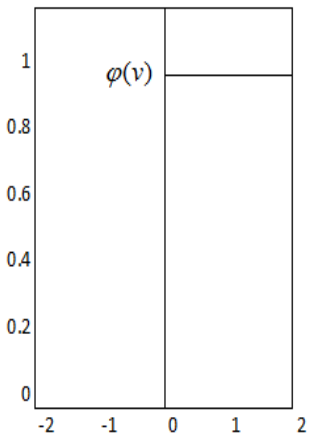
- $x_1\ x_2 \ldots x_m$: signal input
- $w_{k1}\ w_{k2} \ldots w_{km}$: bobot setiap sinapsis k
- u_k : kombinasi linier output yang dihasilkan
- b_k : bias
- φ : fungsi aktivasi
- y_k : signal output neuron
- Pemakaian bias memberi pengaruh terhadap output dari neuron, yaitu :

  $v_k = u_k + b_k$ atau $v_k = \sum_{j=0}^{m} w_{kj} x_j$
- Pada persamaan diatas ditambahkan satu synapsis untuk mengakomodasi term b $X_0 = +1$
- Dan Bobot untuk synapsis ini adalah $W_{k0} = b_k$

Macam2 fungsi aktivasi

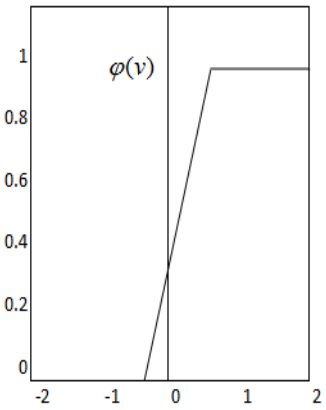
1. Fungsi Treshold, akan menghasilkan 2 output.

$$\varphi(v) = \begin{cases} 1 \text{ if } v \geq 0 \\ 0 \text{ if } v < 0 \end{cases}$$



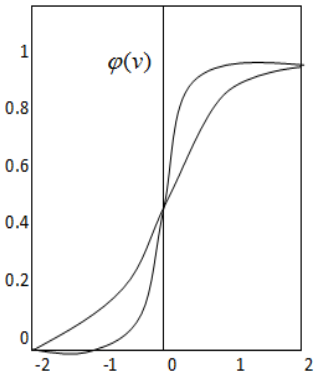
2. Fungsi linier piecewise.

$$\varphi(v) = \begin{cases} 1, & v \geq \frac{1}{2} \\ v + \frac{1}{2}, & -\frac{1}{2} < v < \frac{1}{2} \\ 0, & v \leq -\frac{1}{2} \end{cases}$$



3. Fungsi Sigmoid

$$\varphi(v) = \frac{1}{1 + \exp(-av)}$$



Syarat fungsi dapat menjadi fungsi aktifasi :

1. Nonlinier, dengan fungsi nonlinier akan memperbaiki kemampuan network dalam melakukan tugasnya.
2. Saturate, fungsi yang memiliki nilai minimum dan maksimum. Dengan demikian akan menjaga nilai bobot w dan bias b bounded (terbatas), sehingga waktu training terbatas juga.
3. Kontinuitas dan smoothness, fungsi aktivasi terdefinisi dalam range dari argumenya.

Single layer perceptron

Konsep perceptron yang diajukan oleh Rosenblatt merupakan bentuk desain NN yang paling sederhana yang digunakan untuk mengelompokkan objek dari dua kelas yang bisa dipisahkan secara linier. Perceptron terdiri dari sebuah neuron dengan sinapsis dan bias yang nilainya bisa diatur untuk memperoleh solusi yang tepat. Roosenblatt membultikan bahwa jika data yang digunakan untuk mentraining perceptron tersebut diambil secara random dari set objek dari dua kelas yang bisa dipisahkan secara linier, akan menghasilkan solusi setelah sekian kali iterasi. Dengan menambah jumlah neuron diharapkan bisa mengatasi kasus dimana objek berasal dari multi kelas (terbatas kasus yang linier).

Procedur learning

1) Tujuan proses learning untuk menemukan bobot w dan bias, b atau wq sehingga network secara tepat menghasilkan output {-1,+1} untuk setiap data training yang diinput
2) Salah satu cara untuk melatih perceptron adalah mengawali nilai w dan b dengan nilai random.
3) Selanjutnya secara iteratif (berulang) mempebahari nilainya untuk setiap titik data jika nilainya tidak sesuai ouput yg diinginkan.

1.3. Praktikum Neural Network Menggunakan Excel

1) Buatlah terlebih dahulu tabel keputusan bermain bola berikut menggunakan excel.

DATA KEPUTUSAN MAIN SEPAK BOLA

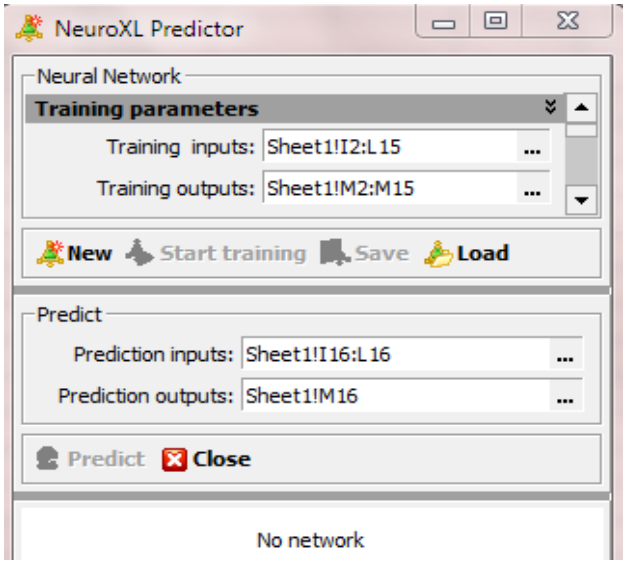
Data	Cuaca	Temp	Kelembaban	Angin	Main
1	Cerah	Panas	Tinggi	Kecil	**Tidak**
2	Cerah	Panas	Tinggi	Besar	**Tidak**
3	Mendung	Panas	Tinggi	Kecil	**Ya**
4	Hujan	Sedang	Tinggi	Kecil	**Ya**
5	Hujan	Dingin	Normal	Kecil	**Ya**
6	Hujan	Dingin	Normal	Besar	**Tidak**
7	Mendung	Dingin	Normal	Besar	**Ya**
8	Cerah	Sedang	Tinggi	Kecil	**Tidak**
9	Cerah	Dingin	Normal	Kecil	**Ya**
10	Hujan	Sedang	Normal	Kecil	**Ya**
11	Cerah	Sedang	Normal	Besar	**Ya**
12	Mendung	Sedang	Tinggi	Besar	**Ya**
13	Mendung	Panas	Normal	Kecil	**Ya**
14	Hujan	Sedang	Tinggi	Besar	**Tidak**
15	Cerah	Dingin	Tinggi	Besar	

2) Rubahlah data tabel diatas menjadi nilai numerik / nominal sebagai berikut :

RUBAH DATA KEPUTUSAN MENJADI NOMINAL

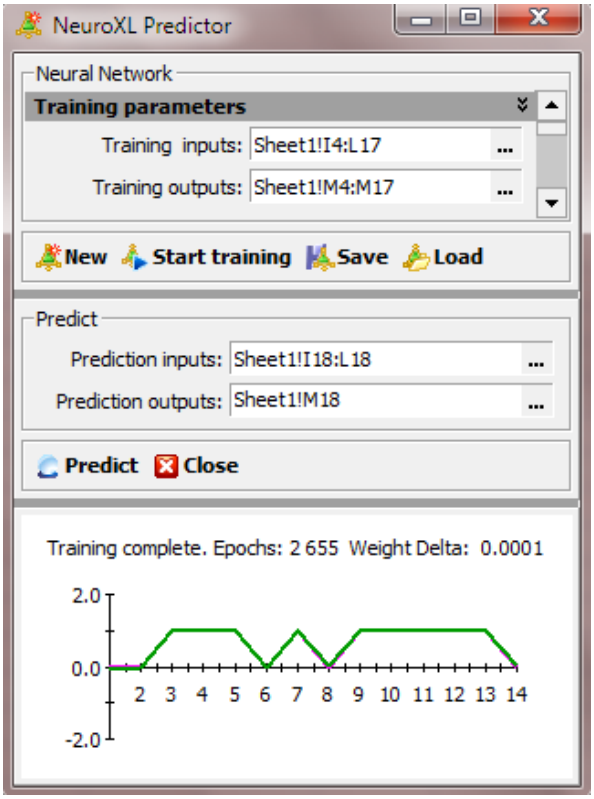
Data	Cuaca	Temp	Kelembaban	Angin	Main
1	1	1	1	1	**0**
2	1	1	1	2	**0**
3	2	1	1	1	**1**
4	3	2	1	1	**1**
5	3	3	2	1	**1**
6	3	3	2	2	**0**
7	2	3	2	2	**1**
8	1	2	1	1	**0**
9	1	3	2	1	**1**
10	3	2	2	1	**1**
11	1	2	2	2	**1**
12	2	2	1	2	**1**
13	2	1	2	1	**1**
14	3	2	1	2	**0**
15	1	3	1	2	

3) Instal add-in neuroxl sebagai fitur tambahan untuk ms excel (NXLPtr.exe)
4) Lakukan analisis menggunakan neural network, melalui menu Add-ins >> PredictorXL >> NeuroXLPredictor, jika sudah aktif akan ditampilkan sebagai berikut :



5) Traning input diisi dengan range cel H4 sampai L17
Training output disisi dengan range cel M4 sampai M17
Predict Input diisi dengan range cel I18 sampai L18
Predict Ouput disi dengan cel M18

Klik button New >> Klik Start Training >> Klik Predict, maka akan ditampilkan pada cel M18 nilai prediksi antara 0 sampai 1, hasil prediksi diperoleh hasil 0,29.



15	1	3	1	2	**0.291031**

Hasil Prediksi

BAB 9
SUPPORT VECTOR MACHINE (SVM)

1.1. Tujuan Praktikum

1. Mahasiswa dapat menggunakan mechine learning support vector machine untuk klasifikasi dan prediksi.
2. Mahasiswa dapat menjelaskan teknik klasifikasi dan prediksi menggunakan algorithma *support vector machine*.

1.2. Pendahuluan

Support Vektor Machine merupakan teknik untuk melakukan prediksi dalam kasus klasifikasi maupun regresi, diajukan oleh Vapnik (1992-1995). SVM berada satu kelas dengan ANN dalam hal fungsi dan kondisi permasalahan yang bisa diselesaikan. Keduanya masuk dalam kelas supervised learning. Permasalahan yang dapat diselesaikan seperti finansial, cuaca dan kodokteran. Dalam implementasinya SVM lebih baik dari ANN terutama dalam solusi yang dicapai. ANN menemukan solusi berupa lokal optimal, sedangkan SVM global optimal. Solusi pada ANN dari setiap training selalu berbeda, hal ini disebabkan solusi lokal optimal yang dicapai tidak selalu sama. Sedangkan SVM selalu mencapai solusi yang sama pada setiap training. Dengan SVM kita akan menemukan fungsi pemisah (klasifier) yang optimal yang bisa memisahkan 2 set data dari dua kelas berbeda.

Kasus klasifikasi linier (Hiperplane) yang bisa dipisahkan, sehingga fungsi pemisah yang dicari adalah fungsi linier. Fungsi ini bisa didefinisikan dengan : g(x)=sgn(f(x)), dengan f(x) W^TX+b, Dimana $w, x \in \Re^n$ dan $b \in \Re$

Klasifikasi bisa dirumuskan sebagai berikut :

1. Kita ingin menemukan parameter (w,b)
2. Kita ingin menentukan fungsi pemisah (klasifier) “terbaik” diantara fungsi yang tidak terbatas jumlahnya untuk memisahkan dua macam objek.
3. Klasifier terbaik adalah yang terletak ditengah-tengah antara dua set objek.

Mencari klasifier terbaik ekuivalen (mirip) dengan memaksimalkan margin atau jarak antara dua set objek dari kelas yang berbeda.

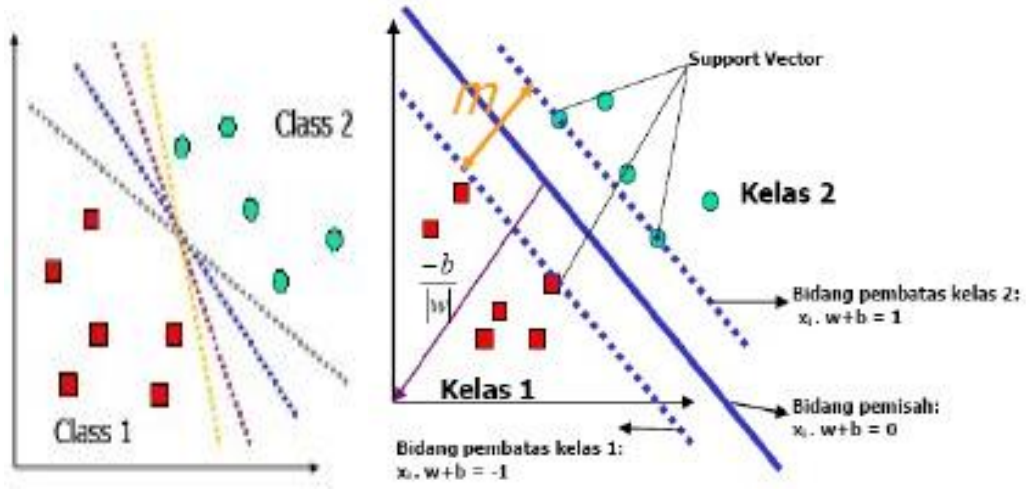
1. Jika $wx_1+b=+1$ adalah klasifier-pendukung dari kelas +1 ($wx_1+b=+1$)
2. Jika $wx_1+b=-1$ adalah klasifier-pendukung dari kelas -1 ($wx_2+b=-1$)
3. Margin antara dua kelas dapat dihitung dengan mencari jarak antara dua klasifier-pendukung, dengan rumus :

$(wx_1+b=+1) - (wx_2+b=-1) \Rightarrow w(x_1-x_2) = 2 \Rightarrow \quad \frac{w}{\| w \|}(x_1 - x_2) = \frac{2}{\| w \|}$

Pada dasarnya jumlah **fungsi pemisah tidak terbatas banyaknya**. Misalnya kita ambil dua fungsi saja yaitu f1(x) dan f2(x). Fungsi f1 mempunyai margin yang lebih besar dari f2. Setelah menemukan dua fungsi ini, data baru masuk dengan keluaran -1. Setelah itu dikelompokkan apakah data baru ada dalam kelas -1 atau +1 menggunakan fungsi pemisah yang sudah kita temukan. Dengan menggunakan f1 kita akan mengelompokkan data baru di kelas -1 yang berarti benar. Dengan menggunakan f2 kita akan menampatkanya dikelas +1 yang berarti salah.

1.3. SVM pada *Linearly Separable Data*

Linearly separable data merupakan data yang dapat dipisahkan secara linier. Misalkan {X1,…,Xn} adalah dataset dan {+1,-1} adalah label kelas dari data Xi. Pada gambar diatas dapat dilihat berbagai alternatif bidang pemisah yang dapat memisahkan semua data set sesuai dengan kelasnya. Namun, bidang pemisah terbaik tidak hanya dapat memisahkan data tetapi juga memiliki margin (m) paling besar.



1.4. SVM pada NonLinearly Separable Data

Untuk mengklasifikasikan data yang tidak dapat dipisahkan secara linier formula SVM harus dimodifikasi karena tidak akan ada solusi yang ditemukan. Oleh karena itu, kedua bidang pembatas harus diubah sehingga lebih fleksibel (untuk kondisi tertentu) dengan penambahan variabel menjadi :

i x.w+b ≥1 − ξ untuk kelas 1

i x.w+b ≤1 + ξ untuk kelas 2.

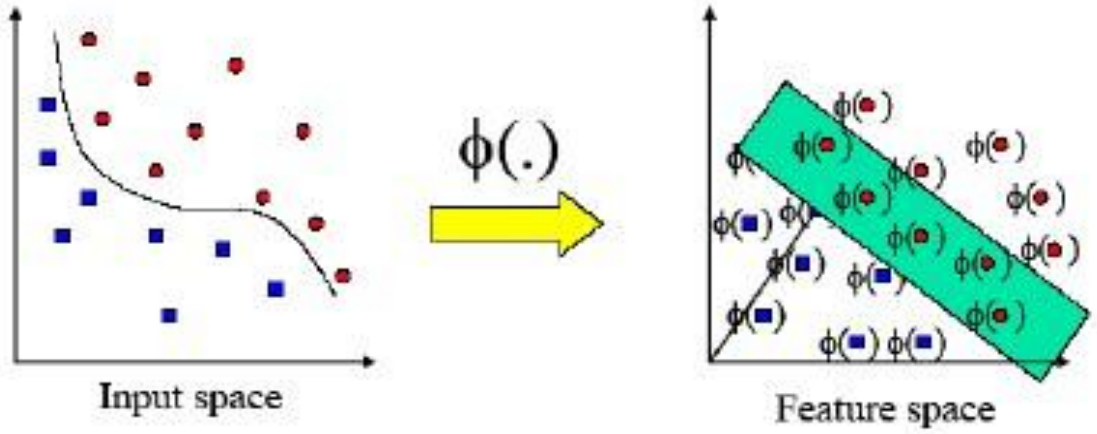
Pencarian bidang pemisah terbaik dengan penambahan variable i ξ sering juga disebut *soft margin hyperplane*. Dengan demikian formula pencarian bidang pemisah terbaik soft hiperplane berubah menjadi :

$$\min \frac{1}{2}|w|^2 + c(\sum_{i=1}^{n} \zeta_1)$$

$$y_1(w.x_1 + b) \geq 1$$

$$\zeta_1 \geq 0$$

c adalah parameter yang menentukan besar penalti akibat kesalahan dalam klasifikasi data dan nilainya ditentukan oleh pengguna. Metode lain untuk mengklasifikasikan data yang tidak dapat dipisahkan secara linier adalah dengan mentransformasikan data ke dalam dimensi ruang fitur (*feature space*) sehingga dapat dipisahkan secara linier pada *feature space*.



1.5. Praktikum Support Vector Machine (SVM) menggunakan WEKA.

1. Buka kembali tabel keputusan bermain bola, dan tambahkan 3 data baru seperti pada tabel berikut untuk klasifikasi dan prediksi.

DATA KEPUTUSAN MAIN SEPAK BOLA

Data	Cuaca	Temp	Kelembaban	Angin	Main
1	Cerah	Panas	Tinggi	Kecil	**Tidak**
2	Cerah	Panas	Tinggi	Besar	**Tidak**
3	Mendung	Panas	Tinggi	Kecil	**Ya**
4	Hujan	Sedang	Tinggi	Kecil	**Ya**
5	Hujan	Dingin	Normal	Kecil	**Ya**
6	Hujan	Dingin	Normal	Besar	**Tidak**
7	Mendung	Dingin	Normal	Besar	**Ya**
8	Cerah	Sedang	Tinggi	Kecil	**Tidak**
9	Cerah	Dingin	Normal	Kecil	**Ya**
10	Hujan	Sedang	Normal	Kecil	**Ya**
11	Cerah	Sedang	Normal	Besar	**Ya**
12	Mendung	Sedang	Tinggi	Besar	**Ya**
13	Mendung	Panas	Normal	Kecil	**Ya**
14	Hujan	Sedang	Tinggi	Besar	**Tidak**
15	Cerah	Dingin	Tinggi	Besar	
16	Cerah	Dingin	Tinggi	Kecil	
17	Mendung	Sedang	Normal	Kecil	

2. Buka file hasil konversi menggunakan arff viewer di Weka *Machine Learning* dan simpan dengan format .arff

ARFF-Viewer - D:\KULIAH GENAP 2011-...

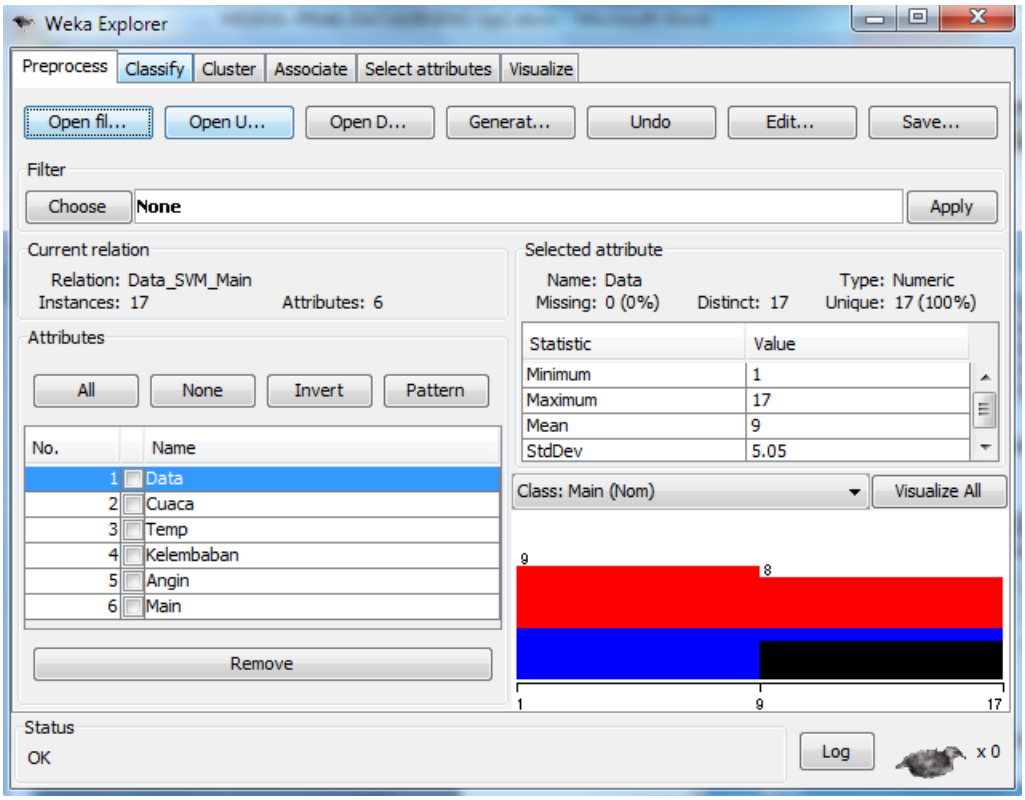
File Edit View

Data_SVM_Main.csv

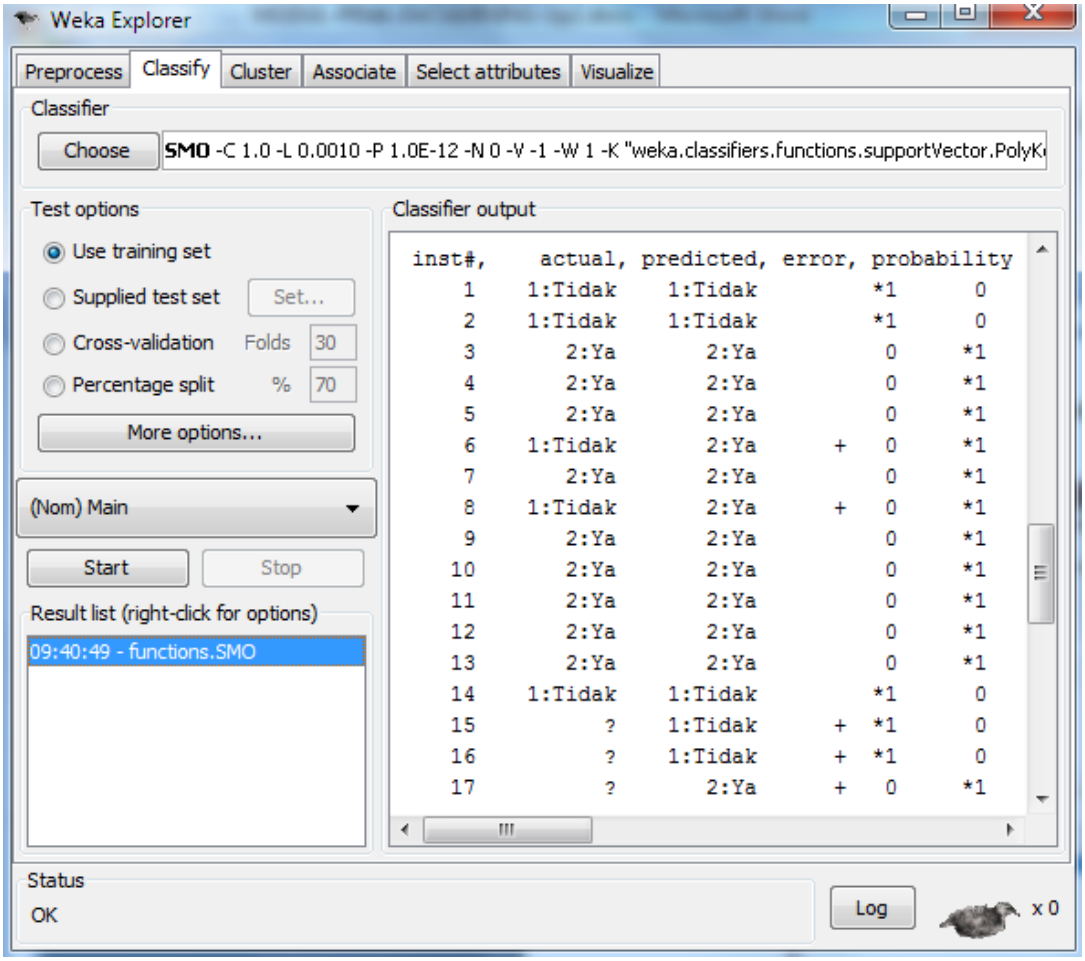
Relation: Data_SVM_Main

No.	Data Numeric	Cuaca Nominal	Temp Nominal	Kelembaban Nominal	Angin Nominal	**Main** Nominal
1	1.0	Cerah	Panas	Tinggi	Kecil	Tidak
2	2.0	Cerah	Panas	Tinggi	Besar	Tidak
3	3.0	Mendung	Panas	Tinggi	Kecil	Ya
4	4.0	Hujan	Sedang	Tinggi	Kecil	Ya
5	5.0	Hujan	Dingin	Normal	Kecil	Ya
6	6.0	Hujan	Dingin	Normal	Besar	Tidak
7	7.0	Mendung	Dingin	Normal	Besar	Ya
8	8.0	Cerah	Sedang	Tinggi	Kecil	Tidak
9	9.0	Cerah	Dingin	Normal	Kecil	Ya
10	10.0	Hujan	Sedang	Normal	Kecil	Ya
11	11.0	Cerah	Sedang	Normal	Besar	Ya
12	12.0	Mendung	Sedang	Tinggi	Besar	Ya
13	13.0	Mendung	Panas	Normal	Kecil	Ya
14	14.0	Hujan	Sedang	Tinggi	Besar	Tidak
15	15.0	Cerah	Dingin	Tinggi	Besar	
16	16.0	Cerah	Dingin	Tinggi	Kecil	
17	17.0	Mendung	Sedang	Normal	Kecil	

3. Buka data pada eksplorer Weka, kemudian pilih tab Classify>> pilih Function >> SMO >> Ok



4. Output hasil klasifikasi dan pengelompokkan yang diperoleh hasil pengolahan dengan weka.



=== Run information ===

Scheme: weka.classifiers.functions.SMO -C 1.0 -L 0.0010 -P 1.0E-12 -N 0 -V -1 -W 1 -K "weka.classifiers.functions.supportVector.PolyKernel -C 250007 -E 1.0"
Relation: Data_SVM_Main
Instances: 17
Attributes: 6
Data
Cuaca
Temp
Kelembaban
Angin
Main
Test mode: evaluate on training data

=== Classifier model (full training set) ===
SMO

Kernel used:
Linear Kernel: K(x,y) = <x,y>

Classifier for classes: Tidak, Ya

BinarySMO

Machine linear: showing attribute weights, not support vectors.

```
       -0.0612 * (normalized) Data
 +      -0.4708 * (normalized) Cuaca=Cerah
 +       0.9587 * (normalized) Cuaca=Mendung
 +      -0.4879 * (normalized) Cuaca=Hujan
 +      -0.3752 * (normalized) Temp=Panas
 +       0.3339 * (normalized) Temp=Sedang
 +       0.0413 * (normalized) Temp=Dingin
 +       1.0413 * (normalized) Kelembaban
 +      -0.6661 * (normalized) Angin
 +       0.425

Number of kernel evaluations: 75 (86.842% cached)

Time taken to build model: 0.01 seconds

=== Predictions on training set ===

 inst#,    actual, predicted, error, probability distribution
     1   1:Tidak   1:Tidak               *1     0
     2   1:Tidak   1:Tidak               *1     0
     3     2:Ya      2:Ya                0    *1
     4     2:Ya      2:Ya                0    *1
     5     2:Ya      2:Ya                0    *1
     6   1:Tidak   2:Ya          +        0    *1
     7     2:Ya      2:Ya                0    *1
     8   1:Tidak   2:Ya          +        0    *1
     9     2:Ya      2:Ya                0    *1
    10     2:Ya      2:Ya                0    *1
    11     2:Ya      2:Ya                0    *1
    12     2:Ya      2:Ya                0    *1
    13     2:Ya      2:Ya                0    *1
    14   1:Tidak   1:Tidak               *1     0
    15        ?    1:Tidak     +         *1     0
    16        ?    1:Tidak     +         *1     0
    17        ?    2:Ya        +         0    *1

=== Evaluation on training set ===
=== Summary ===

Correctly Classified Instances          12      85.7143 %
Incorrectly Classified Instances         2      14.2857 %
Kappa statistic                          0.6585
Mean absolute error                      0.1429
Root mean squared error                  0.378
Relative absolute error                 30.7692 %
Root relative squared error             78.8263 %
Total Number of Instances               14
Ignored Class Unknown Instances          3
```

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.6	0	1	0.6	0.75	0.717	Tidak
	1	0.4	0.818	1	0.9	0.813	Ya
Weighted Avg.	0.857	0.257	0.883	0.857	0.846	0.778	

=== Confusion Matrix ===

```
 a b   <-- classified as
 3 2 | a = Tidak
 0 9 | b = Ya
```

5. Hasil klasifikasi dan predikasi untuk data ke-15,16 menghasilkan “tidak”, sedangkan data ke 17 menghasilkan “ya”. Terdapat 2 data yang tidak terklasifikasi dengan baik yaitu data nomor 6 dan 8 seharusnya terklasifikasi “tidak”, oleh SVM terklasifikasi “ya”.